

Métodos para la Navegación Autónoma Reactiva: Evolución y Tendencias

Hugo Berti
*Facultad de Ingeniería
Universidad Nacional
de La Pampa*
Calle 9 esq. 110, L6360GLH
General Pico, LP, Argentina
hberti@ing.unlpam.edu.ar

Angel D. Sappa
*Centro de Visión por Computador
Universidad Autónoma
de Barcelona*
Edifici O Campus UAB, 08193
Bellaterra, Barcelona, España
angel.sappa@cvc.uab.es

Osvaldo E. Agamennoni
*Dto. de Ing. Eléc. y de Comp.
Universidad Nacional
del Sur*
Av. Alem 1253, B8000CPB
Bahía Blanca, BA, Argentina
oagamen@uns.edu.ar

Resumen—Este trabajo presenta una revisión detallada de los métodos propuestos en la bibliografía para la navegación autónoma de robots móviles. En particular, esta revisión está focalizada en el estudio de los distintos métodos de evitación de colisiones, puesto que constituyen la base para cualquier sistema de navegación autónoma en ambientes dinámicos, parcialmente conocidos o desconocidos. Se presenta, entonces, un resumen cronológico con los métodos más relevantes desarrollados en los últimos años, y se incluyen además las principales evoluciones de los mismos.

Palabras Clave—Robot móvil, navegación autónoma, evitación de colisiones.

I. INTRODUCCIÓN

Las estrategias para el control de movimiento en robots móviles pueden clasificarse como: *métodos de planificación de trayectorias* y *métodos de evitación de colisiones*. Los métodos de planificación utilizan un modelo o mapa del ambiente para calcular una trayectoria desde la posición inicial del robot hasta la *posición objetivo*; mientras que los métodos de evitación de colisiones requieren la información proveniente de los sensores para determinar un movimiento que evitará colisiones con los obstáculos presentes en el ambiente.

En la mayoría de las aplicaciones de robótica móvil el ambiente es dinámico y parcial o completamente desconocido. Bajo estas circunstancias, las trayectorias generadas por los métodos de planificación resultan incorrectas y se requiere una replanificación para alcanzar el objetivo. Además, los métodos de planificación de trayectorias resultan ineficientes para la implementación en tiempo real, por tanto, resultan inadecuados para atender eventualidades durante los movimientos del robot. En tales situaciones, se utilizan estrategias computacionalmente eficientes para conducir al robot en forma segura al objetivo. Estas estrategias, conocidas como *métodos de evitación de colisiones* o *métodos de navegación reactiva*, se nutren de la información proveniente de los sensores para conducir al robot rumbo al objetivo a la vez que se evitan los obstáculos que se presentan en forma imprevista.

En este trabajo se presentan los principales métodos para la navegación autónoma reactiva propuestos en los últimos años. Los métodos presentados en las secciones II, III, y IV se introducen en forma breve debido a su sencillez. Mientras que en las secciones V y VI se describen con mayor detalle métodos cuya aplicación y desarrollo continúa aún vigente.

II. MÉTODO DE NAVEGACIÓN BASADO EN REPRESENTACIONES DE SUPERFICIES

Uno de los primeros desarrollos utilizados en navegación autónoma fue el método de seguimiento de superficies —generalmente paredes— ([1]–[4]). Este método impone al robot un movimiento paralelo a una *superficie de referencia* manteniendo una distancia predefinida. Si a lo largo de la trayectoria

aparece un obstáculo, el robot considera la superficie que define este objeto como la nueva referencia. De esta manera, el robot se desplaza siguiendo el contorno del objeto a la distancia predefinida hasta que pueda restablecer el curso original.

Este método de navegación resulta muy elemental y conveniente sólo en aplicaciones muy específicas. En [5] se utiliza este método para recuperar al robot de un estado de bloqueo o de movimiento cíclico. El concepto de seguimiento de superficies se aplica en [6] para realizar tareas bajo el agua; en este trabajo un vehículo autónomo submarino (AUV)¹ realiza dos tipos de misiones: (i) exploración del fondo del mar y (ii) inspección de cañerías y cables submarinos. En ambas misiones el AUV debe seguir una superficie, fondo del mar o cañería, a una distancia determinada.

III. MÉTODO DE NAVEGACIÓN UTILIZANDO REPRESENTACIONES DE ARISTAS

A diferencia del anterior, un método más general para evitar obstáculos se utilizó en ([7]–[10], entre otros), este método se fundamenta en la representación de los obstáculos contenidos en una escena mediante los bordes que los definen –*aristas*–. De esta manera, un algoritmo basado en la información proveniente de los sensores estima la posición de las aristas e implementa acciones evasivas que permiten conducir al robot rumbo al objetivo por zonas libres de colisiones.

Un ejemplo de aplicación de este método ha sido presentado en [7] donde se implementa esta estrategia de evitación de obstáculos para controlar un robot móvil que brinda asistencia a pacientes tendidos en una cama. En tal trabajo, el robot sólo cuenta con información proveniente de sensores de ultrasonido para la determinación de las aristas de los obstáculos. En el inicio, el sistema realiza una planificación de trayectorias utilizando la información de la posición de los obstáculos estáticos –obstáculos modelados registrados en un *mapa*–. Se considera que los movimientos del robot están limitados a secuencias alternadas de rotaciones y traslaciones puras. Luego, cuando el robot ejecuta una trayectoria y encuentra un obstáculo inesperado, un algoritmo de segmentación extrae las aristas y determina su posición espacial. Seguidamente, utilizando la posición de estas aristas, se emprende una acción evasiva que permite conducir al robot al objetivo evitando la colisión.

Un inconveniente presentado en las primeras implementaciones de este método ([7]–[11]) imponía una detención del robot ante los obstáculos, para luego realizar una secuencia de procesamiento con el fin de evitar colisiones: primero, obtener una vista panorámica de su entorno; luego, procesar la información adquirida para extraer las aristas; y por último, implementar la estrategia de evitación.

IV. MÉTODO DE CAMPO DE POTENCIAL ARTIFICIAL

El método de Campo de Potencial Artificial fue presentado por Khatib en [12] para evitación de colisiones en tiempo real. Manifiesta validez tanto en robots articulados como en robots móviles. Este método ha sido ampliamente utilizado debido a su simplicidad, facilidad de implementación y sobre todo por la gran eficiencia lograda en el control de movimientos en tiempo real. A lo largo de los años se han presentado varias modificaciones a la propuesta original de Khatib, como así también, ejemplos de aplicación del método de campos de potenciales artificiales sobre distintos tipos de robots móviles, algunos ejemplos de ello son: [5], [6], [13]–[16].

El principio de generación de movimiento consiste en plantear un campo de potencial artificial en la escena donde se desplaza el robot. Luego, los movimientos se calculan con la aplicación de:

¹Sigla en inglés correspondiente a *Autonomous Underwater Vehicle*.

(i) una fuerza atractiva generada por la posición objetivo y (ii) fuerzas repulsivas generadas por los obstáculos. La función de potencial artificial definida en una posición $\mathbf{x} = (x, y)$, en la cuál se encuentra el robot, tiene la siguiente expresión

$$U(\mathbf{x}) = U_a(\mathbf{x}) + U_r(\mathbf{x}), \quad (1)$$

donde, $U_a(\mathbf{x})$ es el potencial de atracción producido por la posición deseada en \mathbf{x} , y $U_r(\mathbf{x})$ es el potencial repulsivo inducido por los obstáculos. La fuerza resultante $F(\mathbf{x})$ se expresa como

$$F(\mathbf{x}) = F_a(\mathbf{x}) + F_r(\mathbf{x}), \quad (2)$$

donde,

$$F_a(\mathbf{x}) = -\vec{\nabla} U_a(\mathbf{x}), \quad F_r(\mathbf{x}) = -\vec{\nabla} U_r(\mathbf{x}), \quad (3)$$

aquí, $F_a(\mathbf{x})$ representa la fuerza de atracción que conduce al robot al objetivo y $F_r(\mathbf{x})$ manifiesta la fuerza de repulsión que inducen los obstáculos.

Una vez calculada la fuerza $F(\mathbf{x})$, se consideran las restricciones físicas (velocidad máxima v_{max} y aceleración máxima a_{max}) y la dinámica del robot para establecer los comandos de control de movimiento (velocidad de traslación v y velocidad de rotación w). De esta manera, el robot sometido al campo de potencial artificial $U(\mathbf{x})$ se dirige al objetivo evitando colisionar con los obstáculos presentes en la trayectoria (véase ilustración de la figura 1).

El método de Campos de Potenciales Artificiales se implementó en numerosos sistemas de navegación de robots móviles ([5], [17]–[19], entre otros), y también se utilizó como inspiración de otros métodos de navegación más complejos ([13], [14], [20], [21], entre otros). En [5] se utiliza el concepto de campos de potenciales y grillas de certeza ([22], [23]) para implementar el método de navegación de Campos de Fuerzas Virtuales (VFF)². En [20] se aplica la noción de campos de potenciales y control de impedancia ([24], [25]) para el cálculo de acciones de control evasivas de un robot móvil, considerando además criterios de estabilidad de Lyapunov. En [18] se emplean campos de potenciales artificiales para desarrollar un *controlador difuso* en un robot móvil no-holonómico.

V. MÉTODO DEL HISTOGRAMA DE CAMPO VECTORIAL

El Histograma de Campo Vectorial (VFH)³ es un método de navegación para robots móviles ampliamente utilizado ([6], [26]–[30]). Este método, presentado en [13], permite conducir a un robot móvil rumbo al objetivo al mismo tiempo que se detectan obstáculos desconocidos y se implementan acciones evasivas sin la necesidad de detener al robot. El método está basado en la representación del ambiente mediante mapas constituidos por *grillas de certeza* ([22], [23]) y en el principio de campos de potenciales artificiales ([5], [12]).

En la presente sección se describirá el método original como así también dos sucesivas modificaciones: A) VFH [13], B) VFH+ [14] y C) VFH* [21].

²Sigla en inglés correspondiente a *Virtual Force Field*.

³Sigla en inglés correspondiente a *Vector Field Histogram*.

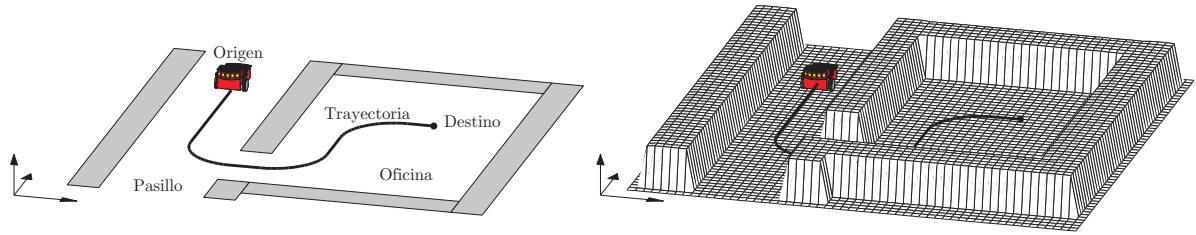


Figura 1. Modelo del ambiente: (*izda.*) elementos de la escena, (*dcha.*) campo de repulsión.

A. Método VFH

La propuesta original del método VFH [13] fue inspirada en brindar solución a un problema inherente al método de campo de potencial artificial. El problema en cuestión se debe a que el método de campo de potencial impone calcular una única fuerza resultante que resume la contribución de todas las fuerzas repulsivas de los obstáculos y la fuerza de atracción de la posición objetivo. La fuerza resultante produce: (*i*) una drástica reducción de la información en situaciones donde se presentan numerosos obstáculos y (*ii*) un resultado erróneo ante situaciones en que los obstáculos se presentan en forma simétrica. Como ejemplo de lo anterior, se puede mencionar el caso en que el robot se encuentra frente a una puerta, en este caso el marco induce fuerzas simétricas en las que se anulan las componentes laterales; en tal situación la fuerza resultante inducirá un retroceso cuando el resultado deseado sería conducir al robot a través de la puerta.

El método VFH soluciona los problemas ocasionados por considerar una única fuerza resultante utilizando un *histograma polar*. Dicho histograma está constituido por sectores angulares obtenidos de dividir los 360 grados del entorno del robot en n secciones angulares (véase ilustración de la figura 2 (*izda.*)). El centro del histograma está definido por la posición del robot y cada sector del histograma contiene información sobre la existencia de obstáculos en dicho sector. Por tanto, analizando el histograma se podrán determinar: *sectores permitidos* –aquellos sectores seguros para el robot– y *sectores prohibidos* –aquellos sectores que presentan riesgo de colisión–. Finalmente, entre los sectores permitidos se selecciona aquel que resulta más conveniente para conducir al robot rumbo al objetivo (véase ilustración de la figura 2 (*dcha.*)).

El cálculo de la dirección que conduce al robot al objetivo sin colisionar con obstáculos consiste de dos etapas. En la primera, se genera el histograma polar basado en un *mapa local*, representado por grillas de certeza, en el cuál se determinan direcciones de movimiento seguras para el robot. En la segunda etapa, se utiliza el histograma polar para determinar la dirección más conveniente que permite conducir al robot rumbo al objetivo sin riesgo de colisión. A continuación se dará un resumen de ambas etapas.

A.1. Primera etapa - Creación del histograma polar: Como se indicó previamente, el método VFH determina las direcciones de movimiento seguras para el robot –direcciones que no presentan riesgo de colisión– utilizando un histograma polar en el cual cada dirección de movimiento tiene asociada una medida de seguridad.

El histograma polar se calcula a partir del ambiente de trabajo del robot; el cuál está representado por un *mapa global* C constituido por celdas, grillas de certeza, que poseen información del estado de ocupación de las mismas. Además del mapa global, se mantiene un mapa local C^* que se desplaza de acuerdo a los movimientos del robot. El mapa C^* , también denominado ventana activa, representa

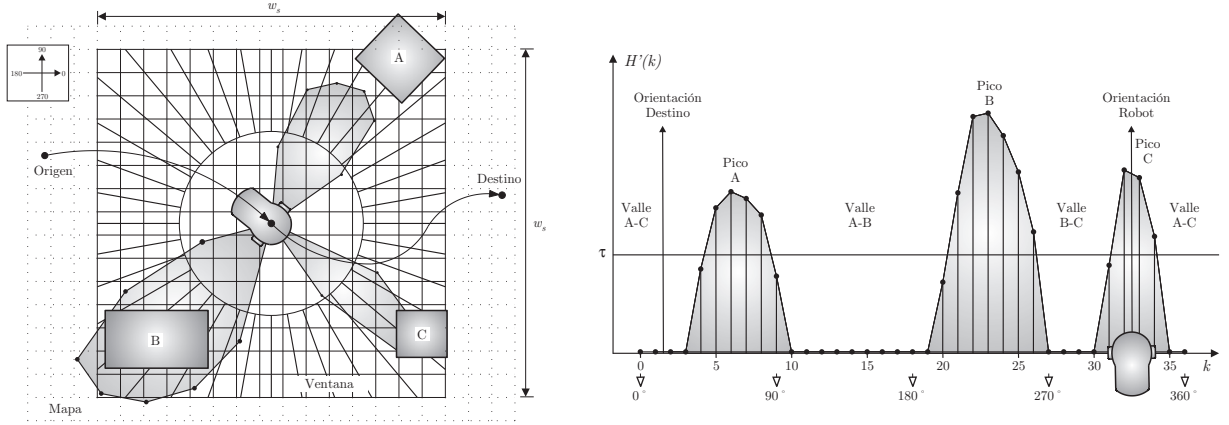


Figura 2. Histograma polar: (*izda.*) versión cartesiana, (*dcha.*) versión polar.

una región cuadrada del mapa global constituida por $w_s \times w_s$ celdas. La celda central está definida por la posición actual del robot (véase figura 2 (*izda.*)). Cada celda de la ventana activa tiene asociado un valor de certeza $c_{i,j}$, que es un índice del estado de ocupación de dicha celda. Inicialmente, cada celda de la ventana activa tiene un valor de certeza $c_{i,j} = 0$, dicho valor se actualiza utilizando la información proveniente de los sensores instalados en el robot (el método original utiliza solamente sensores de ultrasonido). La actualización consiste en incrementar en una unidad el valor de certeza cada vez que se detecta una ocupación en la celda correspondiente [5].

El histograma polar es en un arreglo unidimensional, $H^p = [h_1, \dots, h_k, \dots, h_n]$, que se obtiene de la división de la ventana activa C^* en n sectores polares; dichos sectores tienen un ancho de $\gamma = 360^\circ/n$ (en el método original $n = 72$, con lo cual $\gamma = 5^\circ$). A cada sector polar k se le asigna un valor que representa el estado de ocupación denominado densidad polar de obstáculos h_k y calculado según las celdas activas ($c_{i,j} \geq 1$) que pertenecen a dicho sector

$$h_k = \sum_{c_{i,j} \in C_{i,j}^{*k}} c_{i,j}^2 \cdot (a - b \cdot d_{i,j}) \quad (4)$$

donde, $C_{i,j}^{*k}$ representa el conjunto de celdas contenidas en el sector k ; $d_{i,j}$, la distancia entre $c_{i,j}$ y la celda central de C^* ; a y b , las constantes positivas seleccionadas de modo tal que $a - b d_{max} = 0$; $d_{max} = \sqrt{2} \frac{w_s - 1}{2}$, la distancia máxima desde la celda central a una celda de C^* .

El histograma polar definido por la ecuación (4) presenta discontinuidades debido a la naturaleza discreta de la ventana activa. En [13] se propone aplicar al histograma la siguiente función para suavizar dichas discontinuidades

$$h'_k = \frac{h_{k-l} + 2h_{k-l+1} + \dots + l h_k + \dots + h_{k+l}}{2l + 1} \quad (5)$$

donde, h'_k es la *densidad polar de obstáculos suavizada* y l es un valor definido por el usuario (en el método original $l = 5$). Con los valores h'_k se compone un nuevo histograma, denominado *histograma polar suavizado*, $H'^p = [h'_1, \dots, h'_k, \dots, h'_n]$, en el cuál es posible hallar los sectores permitidos y calcular la dirección que debe seguir el robot, véase ilustración de la figura 2 (*dcha.*).

A.2. Segunda etapa - Determinación de la dirección de movimiento: En esta etapa se utiliza el H'^p y se determina la dirección de movimiento θ_d más conveniente para conducir al robot rumbo al objetivo sin riesgo de colisión. Un histograma polar típico contiene *picos* –sectores consecutivos con alto valor de h'_k – y *valles* –sectores consecutivos con bajo valor de h'_k –. Un valle que está constituido

por sectores con valor h'_k por debajo de un determinado umbral τ se denomina *valle candidato*. La figura 2 (*dcha.*) ilustra el H'_P correspondiente a la escena de la figura 2 (*izda.*), en ella se pueden apreciar 3 valles (**A-B**, **B-C** y **A-C**) y 3 picos (obstáculos **A**, **B** y **C**).

Generalmente, en una escena hay más de un valle candidato; por lo tanto, de todas las posibles opciones el método VFH selecciona aquel valle candidato más cercano a la dirección del objetivo – segmento que une la posición actual del robot con la posición objetivo–. Una vez seleccionado el valle candidato es necesario determinar el sector dentro de dicho valle que definirá la *dirección candidata* –rumbo deseado para el robot–.

En la determinación de la dirección candidata se deben distinguir dos tipos de valles candidatos: *valles anchos* y *valles angostos*. Los valles anchos son aquellos que están constituidos por más de s_{max} sectores consecutivos, en caso contrario, los valles son clasificados como angostos (en el método original $s_{max} = 18$). La diferencia principal entre un valle angosto y un valle ancho radica en el modo en que se definen la direcciones candidatas; en un valle angosto existe sólo una dirección candidata, mientras que en un valle ancho pueden existir de dos a tres direcciones candidatas, entre las que se debe seleccionar una.

En valles angostos la dirección candidata d_c queda definida mediante la siguiente ecuación

$$d_c = \frac{k_{izdo} + k_{dcho}}{2} \quad \text{dirección centrada} \quad (6)$$

donde, k_{izdo} y k_{dcho} son los sectores que se encuentran en los bordes izquierdo y derecho del valle respectivamente.

Por otro lado, en valles anchos se pueden presentar hasta tres direcciones candidatas, dos que se establecen desde cada uno de los laterales del valle y una que se puede presentar cuando la dirección al objetivo se encuentre entre las dos direcciones anteriores. Las direcciones candidatas se determinan según las siguientes ecuaciones

$$\begin{aligned} d_{c,dcha} &= k_{dcho} + \frac{s_{max}}{2}, & \text{hacia la derecha;} \\ d_{c,izda} &= k_{izdo} - \frac{s_{max}}{2}, & \text{hacia la izquierda;} \\ d_{c,\theta_D} &= k_{\theta_D}, & \text{hacia el objetivo, si } k_{\theta_D} \in [d_{c,dcha}, d_{c,izda}]; \end{aligned} \quad (7)$$

donde, k_{θ_D} es el sector que contiene la dirección al objetivo. Entre estas direcciones candidatas, se selecciona aquella que presenta menor diferencia angular respecto del sector orientado al objetivo k_{θ_D}

$$d_c = \begin{cases} d_{c,dcha}, & \text{si } k_{\theta_D} \notin [d_{c,dcha}, d_{c,izda}] \text{ y } A(k_{\theta_D}, d_{c,dcha}) \leq A(k_{\theta_D}, d_{c,izda}); \\ d_{c,izda}, & \text{si } k_{\theta_D} \notin [d_{c,dcha}, d_{c,izda}] \text{ y } A(k_{\theta_D}, d_{c,izda}) < A(k_{\theta_D}, d_{c,dcha}); \\ d_{c,\theta_D}, & \text{si } k_{\theta_D} \in [d_{c,dcha}, d_{c,izda}]; \end{cases} \quad (8)$$

donde, $A(k_i, k_j)$ es el ángulo comprendido entre un sector representado por k_i y otro por k_j .

Finalmente, se calcula la dirección de movimiento θ_d que define el rumbo deseado del robot, permitiendo alcanzar el objetivo y logrando la evitación de obstáculos. La dirección de movimiento se calcula con la siguiente expresión

$$\theta_d = d_c \cdot \gamma \quad (9)$$

La implementación del método VFH proporciona una navegación orientada al objetivo y libre de colisiones; este método fue uno de los primeros que ha posibilitado la evitación de obstáculos sin la necesidad de detener el robot. La representación de un histograma polar evita la drástica reducción de información debida a la posición de obstáculos, y posibilita que se seleccionen direcciones de

movimiento por aberturas estrechas (valles angostos) tales como el pasaje por una puerta: problema típico sin solución si se utilizan únicamente métodos basados en campo de potencial artificial.

B. Método VFH+

El método VFH+, propuesto en [14] y empleado para evitación de obstáculos en robots móviles en tiempo real, ofrece mayores desarrollos sobre el método original VFH [13]. Este último presenta como principales inconvenientes: (i) ignorar explícitamente las dimensiones del robot, (ii) ignorar la dinámica y la cinemática del robot y (iii) evaluar sólo la posición objetivo para la selección de la dirección de movimiento.

En el método original, las dimensiones del robot se consideran en forma implícita en la implementación de un filtro pasa bajos, representado en la ecuación (5), y en la aplicación del valor umbral que define sectores permitidos o prohibidos. La sintonización del filtro pasa bajos (parámetro l en la ecuación 5), como así también la determinación del valor umbral τ , se debe realizar en forma empírica dificultando considerablemente la implementación del método original. Por el contrario, en el método VFH+ se utiliza un filtro pasa bajos determinado en forma teórica; este filtro permite compensar el ancho del robot salvando de este modo uno de los inconvenientes propios de la implementación del método original.

Por otro lado, el método VFH original no contempla ni la dinámica ni la cinemática, calculando la dirección de movimiento analizando sólo la posición objetivo y los obstáculos presentes. A diferencia del anterior, el método VFH+ considera tanto la dinámica como la cinemática del robot por evaluar *trayectorias alcanzables* –trayectorias definidas por pares de velocidades (v, w) que puede desarrollar el robot dentro del próximo ciclo de control–. Las trayectorias alcanzables son consideradas como arcos circulares, representando así una buena aproximación a los movimientos reales desarrollados por una gran variedad de robot móviles ([31]–[35], entre otros).

Asimismo, otro importante desarrollo presentado en el método VFH+ está dado en la utilización de una función de costo que permite la selección de la dirección de movimiento evaluando tres premisas: objetivo, rumbo actual y rumbo calculado previamente. El planteo de una función de costo con premisas permite una navegación dirigida al objetivo, con movimientos suaves y libre de colisiones ([18], [32], [35], entre otros).

El método VFH+ se descompone en cuatro etapas que permiten el planteo de la función de costo para la evaluación y determinación del movimiento que se debe imponer al robot. En las primeras tres etapas, partiendo de una ventana activa de dos dimensiones, se confeccionan sucesivos histogramas que posibilitan: evitar obstáculos, eliminar oscilaciones, y evaluar condiciones cinemáticas y dinámicas. En la etapa final, el método selecciona la dirección de movimiento más conveniente en base al histograma calculado en la tercera etapa y al valor arrojado por la función de costo.

El método VFH+ impone al robot un comportamiento más confiable que su predecesor, logrando una navegación: orientada al objetivo, eficiente, suave y libre de colisiones. En [6] se presenta un sistema de navegación para vehículos autónomos submarinos (AUV), donde el control de movimientos se descompone en dos partes (horizontal y vertical) debido a los sensores y actuadores utilizados. En el control vertical, menos exigente ya que se considera que el AUV siempre tiene la posibilidad de ascender (evitando un bloqueo), se implementa el método VFF [5] basado en campos potenciales. Por el contrario, para el control horizontal se implementa el VFH+ que permite evitar obstáculos y

estados de bloqueos.

C. Método VFH*

El método VFH*, propuesto en [21], es una versión mejorada de los métodos VFH y VFH+, utilizados para evitación de obstáculos de robots móviles en tiempo real. A diferencia de los métodos de evitación de obstáculos puramente locales, el método VFH* realiza un análisis más profundo de las trayectorias candidatas del robot. Este análisis permite detectar en forma anticipada situaciones no deseables, como por ejemplo trayectorias sin salida en las que el robot quedaría atrapado. El análisis de las trayectorias candidatas se realiza utilizando un algoritmo de búsqueda en profundidad A* [36], como así también, funciones de costo y heurísticas, similares a las implementadas por el método VFH+.

En la siguiente sección se resumirán los inconvenientes que generalmente se presentan al implementar algoritmos para evitar obstáculos utilizando información puramente local. Posteriormente, se describirá el método VFH* y sus principales características.

C.1. Problemas de métodos puramente locales: El método VFH+, como así también otros métodos basados en técnicas puramente locales, no funciona adecuadamente en algunas ocasiones al seleccionar la dirección de movimiento. Tales problemas se presentan cuando la información panorámica que dispone el robot en un momento dado no es suficiente para discernir cuál es la trayectoria apropiada para conducir al robot a la posición objetivo. En estos casos, es posible que se seleccione en forma errónea una trayectoria que conducirá al robot por un camino sin salida.

La figura 3(izda.) presenta una ilustración donde se ejemplifica la situación comentada previamente. En este ejemplo, el robot debe avanzar por un pasillo y evitar dos obstáculos que aparecen en su trayectoria en forma consecutiva. El círculo con línea de trazo representa la posición en la que los obstáculos comienzan a tener influencia sobre el control de movimientos del robot. En la situación de la figura, el primer obstáculo comienza a tener influencia sobre los movimientos del robot. Si se implementa el algoritmo VFH+, éste detectará dos valles candidatos: uno que induce a evitar el obstáculo por la izquierda (trayectoria I) y otro que induce a evitar el obstáculo por la derecha (trayectoria II), claramente se aprecia que la trayectoria I conduce al robot a una situación indeseable. Por el contrario, si el robot selecciona la trayectoria II, luego en el punto *p* se puede seleccionar una nueva trayectoria III de modo que el robot pueda alcanzar el objetivo, evitando colisionar con obstáculos y caer en una situación de bloqueo.

Es importante notar que un aumento de la zona de influencia de los obstáculos sobre el control de movimientos del robot no soluciona el problema. Un aumento sólo provocaría que el problema de seleccionar en forma errónea una trayectoria se presente en forma anticipada. Otro inconveniente en el aumento de la zona de influencia de los obstáculos, es que se restringe en forma excesiva las zonas permitidas en ambientes muy poblados, lo cuál puede impedir seleccionar trayectorias dentro de la zona de influencia de los obstáculos que conducen el robot al objetivo.

El problema presentado aquí es común en la mayoría de los algoritmos de evitación de obstáculos basados en técnicas puramente locales. El algoritmo VFH* resuelve este problema planteando una proyección de las posibles trayectorias y analizando sus consecuencias.

C.2. Algoritmo VFH:* Con el fin de introducir el algoritmo VFH*, se recuerda que el método VFH+ construye un histograma polar referido a la posición instantánea del robot; luego, extrae los

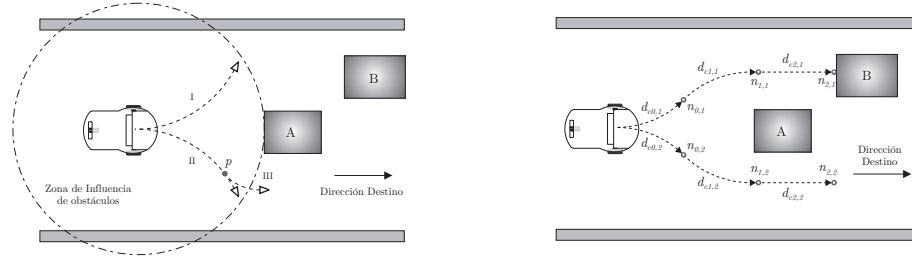


Figura 3. Escenario típico: (*izda.*) situación de ambigüedad, (*dcha.*) evaluación de alternativas.

valles del histograma que presentan trayectorias seguras; posteriormente, determina las direcciones candidatas más convenientes dentro de cada valle. Las direcciones candidatas se evalúan con una función de costo y se selecciona aquella que presenta menor costo para conducir al robot rumbo al objetivo.

A diferencia del método VFH+, el método VFH* analiza la consecuencia de conducir el robot sobre cada dirección candidata –en VFH* llamadas *direcciones candidatas primarias*– antes de realizar la selección definitiva de la dirección de movimiento. Para este propósito, se estiman las nuevas direcciones y orientaciones del robot –*nodos*– que resultarán de conducirlo por cada dirección candidata primaria una distancia d_s . Luego, en cada nodo se utiliza nuevamente el método VFH+ donde: (i) se computa un nuevo histograma polar, (ii) se estiman nuevas direcciones candidatas, en este caso denominadas *direcciones candidatas proyectadas*, y (iii) se calcula un nuevo nodo, como consecuencia de un desplazamiento de longitud d_s por cada dirección candidata proyectada (véase figura 3(*dcha.*)). Este procedimiento se repite n_g veces permitiendo construir un árbol de búsqueda con profundidad n_g , los *nodos finales* corresponden a una distancia total proyectada $d_t = n_g \cdot d_s$.

El objetivo de construir un árbol de búsqueda es calcular la trayectoria proyectada más conveniente. Para ello se plantea una función de costo $g(d_c)$ por cada dirección candidata d_c en forma similar al método VFH+. El costo asociado a cada nodo final se calcula como la suma de los costos de cada uno de los arcos que definen la rama que lo une con el nodo origen. De este modo, la dirección de movimiento que se selecciona es aquella que conduce al robot a un nodo final con menor costo.

El método VFH* emplea un algoritmo de búsqueda A*, la función de costo $g(d_c)$ y una función heurística $h(d_c)$, similar a la función costo del método VFH+, para determinar la dirección candidata primaria d_c más conveniente. El costo asociado a cada nodo se calcula con la siguiente ecuación

$$f(c) = g(d_c) + h(d_c) \quad (10)$$

El procedimiento de expansión de un nodo n_i implementado por el método VFH* consiste de 5 etapas: (i) construcción del histograma polar en la posición de los nodos proyectados (x_i, y_i) , (ii) selección de las direcciones candidatas, (iii) determinación de la posición y orientación de los nodos siguientes n_{i+1} , (iv) cálculo del costo de alcanzar estos nodos, y (v) asignación de sus valores heurísticos [14].

El método VFH* para evitación de obstáculos posibilita una navegación más eficiente, debido a que considera un entorno más amplio que los métodos locales ([13], [14], [32]) para la verificación del progreso de las trayectorias.

VI. MÉTODOS BASADOS EN EL DOMINIO DE VELOCIDADES

Con gran similitud, el *Método de la Curvatura-Velocidad* CVM⁴ [35] y el *Método de la Ventana Dinámica* DWA⁵ [32] fueron desarrollados simultáneamente por grupos de trabajo diferentes. Ambos métodos se distinguen de los anteriores por seleccionar los comandos de control (v, w) , en forma directa, sobre el espacio de velocidades en lugar de hacerlo sobre el *espacio de posiciones* o *espacio de configuraciones* [37]. Los comandos de control se seleccionan de modo tal que maximizan una función objetivo que plantea una situación de compromiso entre navegación: rápida, alejada a los obstáculos y orientada a la meta. El planteo de la función objetivo incorpora las restricciones impuestas por los obstáculos e inherentes del robot en el espacio de velocidades.

Los métodos enunciados, CVM y DWA, coinciden en la selección de los comandos de control (v, w) directamente en el espacio de velocidades. Tal espacio está limitado por restricciones que afectan al comportamiento del robot: aquellas que impone el ambiente (configuración de obstáculos), y aquellas que derivan de las limitaciones físicas del robot (velocidades y aceleraciones máximas). De este modo, el *espacio de búsqueda* de los comandos de control queda reducido por tres tipos de restricciones: (i) trayectorias circulares, (ii) velocidades admisibles, y (iii) velocidades alcanzables; detalladas brevemente a continuación.

Se considera que el robot se desplaza con velocidades constantes (v, w) durante cada ciclo de control ([32], [35], [38]). Por tanto, el robot se mueve a lo largo de *trayectorias circulares* –con curvatura $c = w/v$ – durante dicho ciclo, con las aceleraciones nulas.

Los obstáculos cercanos al robot imponen restricciones sobre las velocidades de traslación y de rotación –*velocidades admisibles*–. Las máximas velocidades admisibles sobre una curvatura dependen de la distancia al siguiente obstáculo a lo largo de dicha curvatura. En la determinación del conjunto de velocidades admisibles, V_a , se utiliza una función, $Dista(v, w)$, que define la distancia al obstáculo más cercano para cada curvatura $c = w/v$, se expresa como

$$Dista(v, w) = \min_{obs \in OBS} dista(v, w, obs), \quad (11)$$

donde obs es un elemento del conjunto de obstáculos OBS . El conjunto V_a se expresa como

$$V_a = \left\{ (v, w) \mid v \leq \sqrt{2 \cdot Dista(v, w) \cdot \dot{v}_{max}}, \quad w \leq \sqrt{2 \cdot Dista(v, w) \cdot \frac{\dot{w}_{max}}{c}} \right\}, \quad (12)$$

donde, \dot{v}_{max} y \dot{w}_{max} son las aceleraciones máximas de traslación y de rotación respectivamente.

Las *velocidades alcanzables* son aquellas que puede adquirir el robot durante el próximo ciclo de control –velocidades que determinan una *ventana dinámica* V_d –. El conjunto V_d se expresa como

$$V_d = \left\{ (v, w) \mid \frac{v - v_a}{\Delta t} \in [-\dot{v}_{max}, \dot{v}_{max}], \quad \frac{w - w_a}{\Delta t} \in [-\dot{w}_{max}, \dot{w}_{max}] \right\}, \quad (13)$$

donde, v_a y w_a son las velocidades de traslación y de rotación actuales del robot respectivamente, y Δt es el intervalo de tiempo del ciclo de control.

A partir de las restricciones presentadas sobre las velocidades del robot, se define el *espacio de búsqueda resultante*, V_r , como

$$V_r = V \cap V_a \cap V_d, \quad (14)$$

⁴Sigla en inglés correspondiente a *Curvature-Velocity Method*.

⁵Sigla en inglés correspondiente a *Dynamic Windows Approach*.

donde V representa el espacio de todas las velocidades posibles del robot, definido por

$$V = \{(v, w) | v \in [0, v_{max}], \quad w \in [-w_{max}, w_{max}]\}. \quad (15)$$

En la figura 4 (*izda.*) se ilustra un ejemplo donde el robot avanza por un pasillo y debe pasar por una puerta a su derecha para alcanzar la meta. En la ilustración de la figura 4 (*dcha.*) se representan los conjuntos de velocidades que intervienen en el cálculo de las acciones de control para la situación de la figura 4 (*izda.*), donde se plantean restricciones adicionales establecidas por robots que emplean conducción diferencial [39].

Una vez determinado el espacio V_r se selecciona una velocidad, dentro de dicho espacio, que maximiza una función objetivo. Dicha función incorpora criterios para conducir al robot rumbo al objetivo, lejos de los obstáculos y a alta velocidad. La función objetivo se expresa como

$$G(v, w) = \sigma(\alpha \cdot Destino(w) + \beta \cdot Distancia(v, w) + \gamma \cdot Velocidad(v)). \quad (16)$$

El cálculo del máximo de la función objetivo en el DWA se realiza por discretización del espacio de búsqueda resultante y por evaluación de cada uno de sus componentes. Mientras que en el CVM se realiza una linealización de todas las funciones y se utilizan métodos de *programación lineal* tradicionales para establecer el valor máximo. A continuación se describen cada uno de los términos de la función objetivo.

A. Función Destino

La función $Destino(w)$ mide la alineación alcanzada por el robot respecto de la meta, α (véase figura 4 (*izda.*), donde se considera que el robot se desplaza con velocidad w durante el intervalo de tiempo del ciclo de control Δt . La figura 5 (A) presenta la evaluación de la función $Destino$ para la situación planteada en la figura 4 (*izda.*). Tanto en el CVM como en el DWA la función destino se calcula de la siguiente manera

$$Destino(w) = 1 - |\alpha - w \cdot \Delta t| / \pi. \quad (17)$$

Esta función presenta el inconveniente de ignorar el acercamiento angular ocasionado por la velocidad de traslación. Este inconveniente manifiesta un error de importancia en los casos que el robot esté próximo y desalineado con la meta. En [40] se plantea una *ley de control ideal* que permite evaluar el acercamiento angular establecido por la velocidad de traslación.

B. Función Distancia

La función $Distancia(v, w)$ representa la distancia al obstáculo más cercano a lo largo de una trayectoria circular cuya curvatura está dada por las velocidades (v, w) –longitud de arco desde el robot a un obstáculo–. Se debe utilizar un valor límite, definido por el usuario, en el caso que no se encuentren obstáculos en una curvatura dada de manera que se pueda establecer una normalización. La figura 5 (B) presenta la evaluación de la función $Distancia$ para la situación planteada en la figura 4. En [34] y [35] se utiliza un algoritmo denominado de *unión mínima* para determinar *funciones lineales a tramos* que indican la distancia de colisión para cada curvatura. En [40] se proponen desarrollos adicionales al DWA que permiten el retroceso del robot, donde se debe calcular la distancia a los obstáculos en trayectorias circulares para tal retroceso.

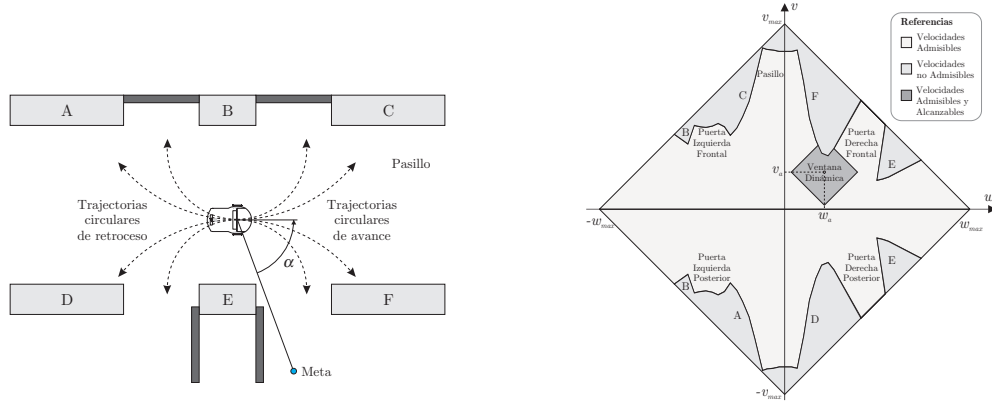


Figura 4. (izda.) Escenario de navegación típico, (dcha.) Restricciones en las velocidades.

C. Función Velocidad

La función *Velocidad* (v) se utiliza para evaluar el progreso del robot sobre la dirección actual de desplazamiento. En el CVM y DWA dicha función se calcula simplemente como una proyección sobre la velocidad de traslación v (véase ilustración de la figura 5 (C))

$$\text{Velocidad}(v, w) = v/v_{max}. \quad (18)$$

Esta función promueve acciones erróneas bajo ciertas circunstancias. Como en el caso que la orientación del robot presente discrepancias considerables respecto de la meta ($\alpha > 90^\circ$), donde se impulsa a un alejamiento de la misma. Por otro lado, la función *Velocidad* no evalúa la cercanía a la meta, por lo tanto cuando el robot se encuentra próximo a ella nuevamente promoverá una acción incorrecta: navegar rápido. En [40] se propone una versión mejorada del método DWA, que utiliza criterios de *estabilidad de Lyapunov* para reducir la velocidad de traslación del robot cuando manifiesta una gran discrepancia angular con el objetivo, e inclusive permite imponer un retroceso cuando la discrepancia angular supera los 90° .

Todos los componentes de la función objetivo, ecuación (16), están normalizados en el intervalo $[0, 1]$ con el fin de poder evaluar y sumar términos con diferentes objetivos. Además, cada término tiene un factor de ponderación (α , β y γ). En la figura 5 (D) se ilustra la evaluación de la función objetivo dentro del espacio de búsqueda –ventana dinámica– para la situación ejemplo de la figura 4 (izda.).

Los métodos basados en el dominio de velocidades han sido muy aceptados por la comunidad científica debido a que permiten una navegación a alta velocidad. En [32] se presentó el método DWA implementándose con el robot móvil RHINO; posteriormente, el mismo grupo de trabajo presentó en [33] nuevos desarrollos al método original denominándolo μ DWA, donde nuevamente el robot RHINO integra los datos de varios tipos de sensores (de ultrasonido y/o láser) y la información del mapa del ambiente para generar movimientos libres de colisiones, la consideración del mapa evita colisionar con objetos no detectados por los sensores utilizados.

Asimismo, en [41] se propone una extensión al DWA denominado *Método de la Ventana Dinámica Global*, que combina planificación de trayectorias con evitación de obstáculos y lo implementa con el robot móvil holonómico NOMAD XR4000. En [38] se presenta otra combinación del método de la ventana dinámica con el método *Step Ahead Dyna-Q⁺* para planificación de trayectorias, en este caso se utilizó el robot móvil PIONEER 1. En [42] se presenta una nueva versión del DWA, denominado

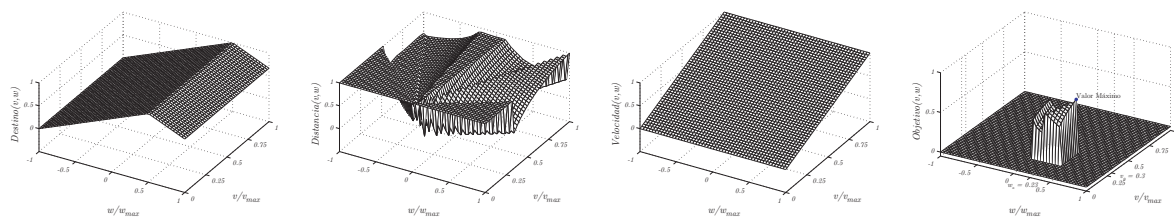


Figura 5. Funciones: (A) *Destino*, (B) *Distancia*, (C) *Velocidad* y (D) *Objetivo*.

Método de la Ventana Dinámica Reducida, que permite la implementación en dos robots móviles no-holónicos: PYGMALION y DONALD. En [34] se presenta el Método de la Senda-Curvatura (LCM)⁶ que plantea una combinación entre métodos basados en el espacio de configuraciones y en el espacio de velocidades. En [43] se presenta un método para la navegación de vehículos de exteriores que aplica el concepto de campo de potencial artificial en el espacio de velocidades.

VII. CONCLUSIONES

En este trabajo se realizó un estudio de los métodos de navegación más importantes propuestos en los últimos años. Se debe destacar que no existe un método de navegación para uso general, que resulte conveniente para todo propósito, en todo ambiente y para todo robot. En cada aplicación en particular se debe analizar el objetivo y el ambiente de operación, luego determinar el robot y sus sensores, y por último se deben establecer los algoritmos de control de movimiento (planificación de tareas, planificación de trayectorias, evitación de obstáculos, control de bajo nivel, comunicación con el robot, entre otros).

El método de navegación basado en representaciones de superficies es muy elemental y resulta conveniente en tareas muy específicas, tales como: limpieza de grandes pasillos en edificios, recuperación de estado de bloqueo, exploraciones del fondo del mar, inspección de cañerías y cables submarinos, etc.

De igual manera, el método de navegación que utiliza representaciones de aristas posee una utilización restringida a aplicaciones muy particulares. En general, este método impone detener el robot frente a los obstáculos inesperados, obtener una imagen panorámica del entorno, extraer las aristas de los obstáculos y finalmente implementar la estrategia de evitación.

El método del campo de potencial artificial es ampliamente utilizado, muy simple, fácil de implementar y conveniente para utilizar en tiempo real. Sin embargo, presenta como desventaja que no es apropiado en ambientes muy poblados ya que el cálculo de una única fuerza repulsiva es una excesiva reducción de información de los obstáculos presentes en una escena.

El método del histograma de campo vectorial original es muy popular. Su principal aporte radica en que puede utilizarse en ambientes muy poblados debido a que divide el espacio de trabajo en sectores polares, de los cuales puede analizarse cual es el sector polar más conveniente para conducir el robot rumbo al objetivo sin riesgo de colisión. La principal desventaja de este método es que no considera la dinámica del robot.

El método de la curvatura velocidad y el método de la ventana dinámica son similares. El principal aporte de ambos es que consideran la dinámica del robot y que seleccionan los comandos de movimientos en el espacio de velocidades. La mayor desventaja de estos métodos, al igual que otros

⁶Sigla en inglés correspondiente a *Lane-Curvature Method*.

basados en el espacio local, es que puede presentar comportamientos cíclicos o alcanzar mínimos locales.

Finalmente, podemos concluir que la tendencia actual es el desarrollo de métodos de evitación de obstáculos bajo las siguientes premisas: (i) plantear una función objetivo, (ii) considerar la dinámica del robot, y (iii) operar en el espacio de velocidades en forma directa o indirecta.

REFERENCIAS

- [1] G. Bauzil, M. Briot y P. Ribes. A navigation sub-system using ultrasonic sensors for the mobile robot HILARIE. En: *1st International Conference on Robot Vision and Sensory Controls*. pp. 47–58. 1981.
- [2] J. L. Crowley. World modeling and position estimation for a mobile robot using ultrasonic ranging. En: *Proceedings of the IEEE Int. Conference on Robotics and Automation (ICRA'89)*. pp. 365–393. 1989.
- [3] G. Giralt. Mobile robots. *NATO ASI Series, Vol. F11, Robotics and Artificial Intelligence* pp. 365–393. 1984.
- [4] J. Iijima, S. Yuma y Y. Kanayama. Elementary functions of a self-contained robot, YAMABICO 3.1. En: *Proceedings of the 11th International Symposium on Industrial Robots*. pp. 211–218. 1983.
- [5] J. Borenstein y Y. Koren. Real-time obstacle avoidance for fast mobile robots. *IEEE Transactions on Systems, Man, and Cybernetics* **19**(5), 1179–1187. 1989.
- [6] G. Antonelli, S. Chiaverini, R. Finotello y R. Schiavon. Real-time path planning and obstacle avoidance for RAIS: An autonomous underwater vehicle. In *IEEE Journal of Oceanic Engineering*. **26**(2), 216–227. 2001.
- [7] J. Borenstein y Y. Koren. Obstacle avoidance with ultrasonic sensors. *IEEE Journal of Robotics and Automation* pp. 213–218. 1988.
- [8] A. Elfes. Sonar-based real-world mapping and navigation. *IEEE Journal of Robotics and Automation* **RA-3**(3), 249–265. 1987.
- [9] R. Kuc y B. Barshan. Navigating vehicles through an unstructures environment. En: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'89)*. pp. 1422–1426. 1989.
- [10] V. Lumelsky y T. Skewis. A paradigm for incorporating vision in the robot navigation function. En: *Proceedings of the IEEE Int. Conference on Robotics and Automation (ICRA'88)*. pp. 734–739. 1988.
- [11] A. M. Flynn. Combining sonar and infrared sensors for mobile robot navigation. *The International Journal of Robotics Research* **7**(6), 5–14. 1988.
- [12] O. Khatib. Real-time obstacle avoidance for manipulators and mobile robots. *The International Journal of Robotics Research* **5**(1), 90–98. 1986.
- [13] J. Borenstein y Y. Koren. The vector field histogram - fast obstacle avoidance for mobile robots. *IEEE Transactions on Robotics and Automation* **7**(3), 278–288. 1991
- [14] I. Ulrich y J. Borenstein. VFH+: Reliable obstacle avoidance for fast mobile robots. En: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'98)*. pp. 1572–1577. 1998.
- [15] C. W. Warren. A technique for autonomous underwater vehicle route planning. *IEEE Journal of Oceanic Engineering* **15**(3), 199–204. 1990.
- [16] D. R. Yoerger, A. M. Bradley, B. B. Walden, H. Singh y R. Bach-mayer. Surveying a subsea lava flow using the autonomous bentic ex-plorer (ABE). En: *Pre-Proc. 6th Int. Advanced Robotics Program, 1996*. pp. 1–21. 1996.
- [17] C. Connolly, J. Burns y R. Weiss. Path planning using Laplace's equation. En: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'90)*. pp. 2102–2106. 1990.
- [18] J. Urzelai, J.P. Uribe y J.M. Ezkerra. Fuzzy controller for wall following with a non-holonomous mobile robot. En: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'97)*. pp. 2505–2511. 1997.
- [19] P. Peussa y A. Virtanen. A general purpose obstacle voidance system. En: *Proceedings of 2nd Tampere International Conference on Machine Automation (ICMA'98)*. pp. 211–220. 1998.
- [20] H. Secchi, R. Carelli y V. Mut. Design of stable algorithms for mobile robot control with obstacle avoidance. En: *Proceedings of 14th World Congress on Automatic Control (IFAC'99)*. pp. 185–190. 1999.
- [21] I. Ulrich y J. Borenstein. VFH*: Local obstacle avoidance with look-ahead verification. En: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'00)*. pp. 2505–2511. 2000.
- [22] A. Elfes. Using occupancy grids for mobile robot perception and navigation. *Computer Magazine* pp. 46–57. 1989.
- [23] H. Moravec. Certainty grids for sensor fusion in mobile robots. *Sensor Devices and Systems for Robotics* pp. 243–276. 1989.

- [24] V. Mut, R. Carelli y B. Kuchen. Adaptive impedance control for robots with sensorial feedback. En: *Proceedings of the XIII National Symposium of Automatic Control (AADECA'92)*. 1992.
- [25] H. Secchi, V. Mut y R. Carelli. Impedance control for mobile robots with sensorial feedback. En: *Proceedings of the XIV National Symposium of Automatic Control (AADECA'94)*. pp. 19–24. 1994.
- [26] E. Huber y D. Kortenkamp. Using stereo vision to pursue moving agents with a mobile robot. En: *Proceedings of the IEEE Int. Conference on Robotics and Automation (ICRA'95)*. pp. 365–393. 1995.
- [27] D. Kortenkamp, M. Huber, F. Koss, W. Belding, J. Lee, A. Wu, C. Bidlack y S. Rogers. Mobile robot exploration and navigation of indoor spaces using sonar and vision. En: *Conference on Intelligent Robotics in Field, Factory, Service, and Space (CIRFFSS)*. Houston, TX, USA. pp. 509–519. 1994.
- [28] Y. D. Kwon y J. S. Lee. An obstacle avoidance algorithm for mobile robot: the improved weighted safety vector field method. En: *Proceeding of the IEEE 10th International Symposium on Intelligent Control*, 1995. pp. 441–446. 1995.
- [29] A. C. Schultz, W. Adams, B. Yamauchi y M. Jones. Unifying exploration, localization, navigation and planning through a common representation. En: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'99)*. pp. 2651–2658. 1999.
- [30] A. Schultz, W. Adams y B. Yamauchi. Integrating exploration, localization, navigation and planning with a common representation. *Autonomous Robots* 6(3), 293–308. 1999.
- [31] W. Feiten, R. Bauer y G. Lawitzky. Robust obstacle avoidance in unknown and cramped environment. En: *Proceedings of the IEEE Int. Conference on Robotics and Automation (ICRA'94)*. pp. 2412–2417. 1994.
- [32] Dieter Fox, Wolfram Burgard y Sebastian Thrun. The dynamic window approach to collision avoidance. *IEEE Robotics and Automation Magazine* 4(1), 23–33. 1997.
- [33] Dieter Fox, Wolfram Burgard y Sebastian Thrun. A hybrid collision avoidance method for mobile robots. En: *Proceedings of the IEEE Int. Conference on Robotics and Automation (ICRA'98)*. Vol. 2. pp. 1238–1243. 1998.
- [34] N. Y. Ko y R. G. Simmons. The lane-curvature method for local obstacle avoidance. En: *IEEE/RSJ Int. Conference on Intelligent Robots and Systems (IROS'98)*. Vol. 3. Victoria, B. C., Canada. 1998.
- [35] R. Simmons. The curvature-velocity method for local obstacle avoidance. En: *Proceedings of the IEEE Int. Conference on Robotics and Automation (ICRA'96)*. Los Alamitos, CA, USA. pp. 3375–3382. 1996.
- [36] S. J. Russell y P. Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall; 2nd edition. ISBN: 0137903952. December 20, 2002.
- [37] J.-C. Latombe. *Robot Motion Planning*. Kluwer Academic Publishers. Boston, MA, USA. ISBN: 0792391292. 1991.
- [38] O. A. A. Orqueda, H. Berti y O. E. Agamennoni. A strategy for safe goal-directed autonomous navigation. En: *Proceedings of the 4th IFAC International Symposium on Intelligent Components and Instruments for Control Applications (SICICA 2000)*. Buenos Aires, Argentina. pp. 135–140. 2000.
- [39] H. Berti, A. D. Sappa y O. E. Agamennoni. Navegación autónoma reactiva en robots móviles con conducción diferencial. En: *Anales XIX Congreso Argentino de Control Automático (AADECA 2004)*. Buenos Aires, Argentina. ISBN: 950-99994-3-1. 2004.
- [40] H. Berti, A. D. Sappa y O. E. Agamennoni. Ventana dinámica mejorada (VDM): Un método para navegación autónoma. En: *III Jornadas Argentinas de Robótica*. Número 16 en: 1. San Juan, Argentina. 2004.
- [41] O. Brock y O. Khatib. High-speed navigation using the global dynamic window approach. En: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'99)*. pp. 341–346. 1999.
- [42] K. O. Arras, J. Persson, N. Tomatis y R. Siegwart. Real-time obstacle avoidance for polygonal robots with a reduced dynamic window. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'02)*. pp. 3050–3055. 2002.
- [43] S. Shimoda, Y. Kuroda y K. Iagnemma. Potential field navigation of high speed unmanned ground vehicles on uneven terrain. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'05)*. pp. 2839–2844. 2005.
- [44] K. Kant y W. Zucker. Toward efficient trajectory planning: The path-velocity decomposition. En: *The International Journal of Robotics Research*. Vol. 5. pp. 72–89. 1986.
- [45] M. Khatib y R. Chatila. An extended potential field approach for mobile robot sensor-based motions. *Intelligent Autonomous Systems (IAS'4)* pp. 490–496. IOS Press. 1995.
- [46] S. Lacroix, S. Fleury, H. Haddad, M. Khatib, F. Ingrad, G. Bauzil, M. Herrb, C. Lemaire y R. Chatila. Reactive navigation in outdoor environments. En: *The International Journal of Robotics Research Special Issue on Field and Service Robotics*. Also LAAS Research Report number 98364. 1998.
- [47] O. Khatib y J. F. Le Maitre. Dynamic control of manipulators operating in a complex environment. En: *Proceedings 3rd CISM-IFTOMM Symp. Theory Practice Robots Manipulators*. pp. 267–282. 1978.