

Formación de Robots Móviles y Evasión de Obstáculos Utilizando Algoritmos PRM y Dijkstra

H. Pérez-Villeda^{*}, F. Mirelez-Delgado, A. Morales-Díaz

Centro de Investigación y Estudios Avanzados del Instituto Politécnico
Nacional, Unidad Saltillo, Ramos Arizpe, Coah. 25900, México Tel:
(844) 4389600 Ext. 8600;

e-mails:

hecvilleda@gmail.com, {flabio.mirelez, america.morales}@cinvestav.edu.mx.

Resumen: La navegación con evasión de obstáculos para un grupo de robots móviles diferenciales es un problema complejo debido a las restricciones no-holónomas de este tipo de sistemas. Se presenta una estrategia de formación virtual para un grupo de robots móviles diferenciales utilizando los algoritmos PRM y Dijkstra para evasión de obstáculos. El grupo de robots es modelado como un robot planar de 3 GDL donde cada unión del robot planar es representada por un móvil diferencial. La plataforma experimental consta de una cámara en cielo para lectura de posición y orientación de los robots, una computadora de escritorio para cálculo centralizado y comunicación Bluetooth con los robots para envío de comandos de velocidad. Se presentan los resultados de la trayectoria admisible que lleva al grupo de robots de una configuración inicial a una configuración final evadiendo obstáculos estáticos y manteniendo la estructura de la formación.

Palabras Clave: robot móvil diferencial; control no lineal; evasión de obstáculos.

1. INTRODUCCIÓN

Los robots móviles diferenciales son un caso especial de estudio ya que son sistemas no holónomos, es decir, que tienen más grados de libertad (GDL) que entradas de control. Este tipo de sistemas son ampliamente usados en la industria como robots seguidores de líneas para transportar materias primas o productos terminados.

Uno de los principales retos para este tipo de sistemas es la planificación de trayectorias en el espacio Cartesiano, ya que a diferencia de los robots móviles omnidireccionales, el movimiento lateral queda restringido y se complica cuando se tienen dos o más robots diferenciales en modo cooperativo. Al tener dos o más robots se aumentan la maniobrabilidad y destreza del sistema, es por esto que en los últimos años, el uso de robots cooperativos es un tema que ha sido desarrollado en la comunidad científica.

En la literatura de robótica móvil se tienen tres esquemas de configuración para el desarrollo de tareas cooperativas: *Líder-seguidor*, en este tipo de configuración uno o más robots son definidos como líderes y el resto del equipo como seguidores restringidos a imitar el comportamiento de sus guías. *Basado en comportamiento*, en este esquema cada integrante tiene un comportamiento predefinido de tal manera que el comportamiento general del grupo depende de las ponderaciones individuales. *Estructura virtual*, los robots que son controlados bajo este concepto están sujetos a permanecer en una estructura o configuración predefinida cumpliendo con modificaciones de forma y posición. En este artículo el grupo de robots móviles diferenciales

será controlado utilizando el enfoque de estructura virtual, para ello cada móvil diferencial representará una unión de dos eslabones en un robot planar virtual de 3 GDL. Usando la configuración de robot planar se pueden calcular las rutas para pasar de una configuración inicial de los robots a una final sin colisiones entre los elementos del grupo o con obstáculos del entorno.

La estructura del artículo es la siguiente: En la sección 2 la formación del grupo de robots y el modelo cinemático de cada robot individual son presentados. La sección 3 explica la obtención de las trayectorias admisibles libres de colisión para el grupo de robots. La sección 4 muestra la formulación del control que garantiza que el grupo de robots siga la trayectoria admisible. La sección 5 describe la obtención de las trayectorias Cartesianas de referencia a partir de la sucesión de configuraciones obtenidas en la sección 3. Finalmente las secciones 6 y 7 contienen los experimentos realizados y las conclusiones de este trabajo respectivamente.

2. FORMACIÓN Y MODELO DEL GRUPO DE ROBOTS MÓVILES

La estructura de la formación de los robots móviles se modela como de un robot planar de 3 GDL, ver Figura 1. Dicha formación está sujeta a las configuraciones posibles que puede adoptar el robot planar, cada robot móvil diferencial (círculos verdes) representa una unión de eslabones virtuales, (l_1 , l_2 , l_3) que son las distancias constantes que se deben de respetar entre los robots, el robot 0 (círculo gris), es un robot virtual que representa la primera articulación del robot planar, finalmente los recuadros rojos representan obstáculos en el ambiente.

^{*} Este trabajo es gracias a CONACyT.

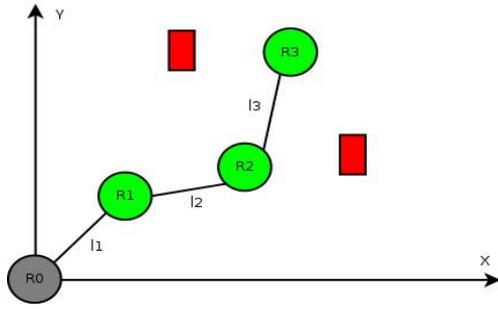


Figura 1. Esquema de un grupo de robots móviles formando un robot planar virtual.

La configuración Cartesiana $\xi = [x, y, \theta]^T$ de cada uno de los robot móvil diferencial de dos ruedas con centro de masa en medio de su eje y con una rueda pasiva puede apreciarse en la figura 2, donde ξ son las coordenadas generalizadas Cartesianas, (x, y) es la posición del centro de masa del robot en el plano y θ denota el ángulo de orientación, véase Figura 2.

El modelo cinemático de uno de los robot móviles en coordenadas Cartesianas está dado por:

$$\dot{x} = v \cos(\theta), \quad \dot{y} = v \sin(\theta), \quad \dot{\theta} = \omega \quad (1)$$

Donde v es la velocidad lineal del robot móvil y ω la velocidad angular, además se considera la restricción no-holónoma $\dot{x} \sin(\theta) - \dot{y} \cos(\theta) = 0$.

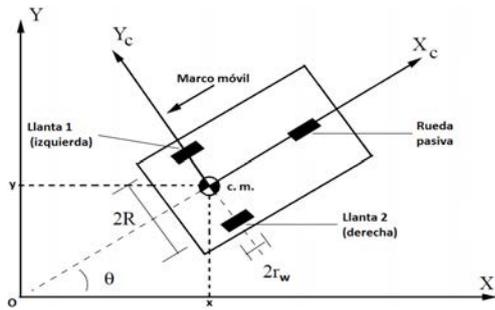


Figura 2. Esquema de un robot móvil en el plano.

3. GENERACIÓN DE TRAYECTORIAS ADMISIBLES

La obtención de una trayectoria admisible libre de colisión que lleve a la formación de robots de una configuración inicial a una configuración final puede lograrse utilizando algoritmos probabilísticos y teoría de grafos.

3.1 Algoritmo Mapa de Caminos Probabilísticos

El movimiento del robot planar virtual esta gobernado por el siguiente vector de configuración:

$$q_{rp} = [q_0, q_1, q_2]^T \quad (2)$$

Debido a que cada robot móvil simula una articulación del robot planar virtual, es posible generar trayectorias para el robot planar que a su vez puedan ejecutarse por el conjunto de robots móviles. El algoritmo mapa de caminos probabilísticos¹ tiene como entradas un par de configuraciones (inicial y final), en este caso, las de un robot planar, así como la geometría de los eslabones y la geometría y posición de los obstáculos. Los argumentos de salida del

¹ PRM por sus siglas en Inglés

algoritmo PRM son un conjunto de configuraciones libres de colisión con estructura de grafo.

La metodología del PRM consiste en construir un grafo que represente la conectividad de varias configuraciones dentro de un espacio libre de colisiones, para ello es necesario muestrear el espacio de configuraciones de manera aleatoria, posteriormente se retienen aquellas que sean factibles (libres de colisión), posteriormente son conectadas con las muestras vecinas mediante un planificador local, finalmente se descartan los caminos que se encuentren en colisión con obstáculos (ver Tabla 1).

Las configuraciones inicial y final se añaden como nuevos nodos al grafo y se procede con la fase de consulta para encontrar el camino más corto entre las últimas configuraciones añadidas al grafo. La convergencia del algoritmo depende directamente del número de muestras aleatorias que se generen, incrementar el número de muestras implica tener un grafo mas grande, lo cual incrementa la probabilidad de encontrar una conexión entre la configuración inicial y la configuración final, sin embargo, si de antemano no existe alguna conexión entre la configuración inicial y la configuración final no importa el número de muestras aleatorias ya que nunca se podrá unir la configuración inicial con la configuración final.

Tabla 1. Algoritmo PRM.

Algoritmo: Construcción del roadmap.

Entrada: n es el número de nodos en G . k es el número de vecinos más próximos por cada configuración.

Salida: El roadmap $G = (V, E)$.

```

V ← 0
E ← 0
Mientras |V| < n hacer
    repetir
        q ← muestra()
        hasta estaLibreColision(q)
        V ← V ∪ {q}
Para toda q ∈ V hacer
    Nq ← el k-ésimo vecino más cercano de q de V usando
    dist
    for all q' ∈ Nq hacer
        si (q, q') ∉ E y caminoLocal(q, q') ≠ NIL
        entonces
            E ← E ∪ {(q, q')}
```

3.2 Algoritmo de Dijkstra

El camino más corto entre dos puntos es la ruta con impedancia mínima, tal impedancia puede ser distancia, tiempo, transporte, tráfico, etcétera. El problema del camino más corto fue resuelto por E. W. Dijkstra en 1959. Actualmente este algoritmo sigue siendo el mas usado para resolver este problema. El algoritmo requiere que el peso de cualquier arista no sea negativo, para asegurar que no haya ningún bucle negativo, haciendo que el problema siempre tenga solución.

El algoritmo de Dijkstra brinda un camino cuya longitud se incrementa gradualmente de acuerdo al orden de la ruta más corta generada nodo a nodo. La idea subyacente en este algoritmo consiste en ir explorando todos los caminos que parten del vértice origen y que llevan a todos los demás vértices; cuando se obtiene el camino más corto desde el vértice origen al resto de vértices que componen el grafo, el algoritmo se detiene (ver Tabla 2).

Tabla 2. Algoritmo Dijkstra.

Algoritmo: Búsqueda de las rutas más cortas.

Entrada: un grafo dirigido $G = (V, E)$, el nodo inicial $v_0 \in V$, la función de aristas a reales positivos con $l(v_i, v_j) = \infty$ si (v_i, v_j) no es una arista, $v_i \neq v_j$ y $l(v, v) = 0$.

Salida: Para cada nodo $v \in V$, la ruta más corta de v_0 a v con respecto a todas las posibles de acuerdo a la suma de las etiquetas de las aristas.

$S \leftarrow \{v_0\}$
 $D[v_0] \leftarrow 0$
 Para cada v en $V - \{v_0\}$ hacer $D[v] \leftarrow l(v_0, v)$
 Mientras $S \neq V$ hacer
 tomar un nodo w en $V - S$ tal que $D[w]$ sea mínimo
 agregar w a S
 Para cada v en $V - S$ hacer
 $D[v] \leftarrow \min(D[v], D[w] + l(w, v))$

La sucesión de configuraciones admisibles libres de colisión obtenidas por el algoritmo Dijkstra, que lleva al grupo de robots de una configuración inicial a una configuración final sin colisionar con los obstáculos puede expresarse como:

$$q_i = [q_{i0}, q_{i1}, q_{i2}, \dots, q_{in}]^T \quad i = 0, 1, 2, 3 \quad (3)$$

En donde:

q_i representa a la sucesión de configuraciones que debe de adoptar el i -ésimo robot, la cual esta conformada por q_{i0} que representa la configuración inicial del i -ésimo robot, q_{in} que representa la configuración final del i -ésimo robot y $q_{i1}, q_{i2}, \dots, q_{i(n-1)}$ que representa la sucesión de configuraciones que une a q_{i0} con q_{in} . El algoritmo Dijkstra asegura que la sucesión $q_{i1}, q_{i2}, \dots, q_{i(n-1)}$ sea la trayectoria más corta del conjunto de configuraciones muestreado que une a la configuración q_{i0} con q_{in} .

4. CONTROL

Como se muestra en Kostić *et al.* [2010] y Sadowska *et al.* [2011], se considera un grupo de robots móviles diferenciales que deben mantener una formación deseada mientras que cada robot sigue una trayectoria de referencia. Para lograr este fin se considera un control donde todos los robots siguen su propia trayectoria considerando la restricción no holónoma de esta clase de sistemas. El acoplamiento entre los robots se define considerando las coordenadas Cartesianas para desarrollar diferentes trayectorias. Considerando el modelo cinemático presentado en la sección 2, el modelo para un grupo de n robots móviles diferenciales queda dado por las siguientes ecuaciones Kanayama *et al.* [1990]:

$$\dot{x}_i = v_i \cos \theta_i \quad \dot{y}_i = v_i \sin \theta_i \quad \dot{\theta}_i = \omega_i \quad (4)$$

Para $i = 1, \dots, n$, donde x_i y y_i son las coordenadas Cartesianas del i -ésimo robot, v_i y ω_i son las velocidades lineal y angular, respectivamente, y θ_i es el ángulo de orientación respecto al eje X del plano de referencia. De acuerdo con el plano de referencia y considerando la postura deseada (véase la Figura 3), el error de seguimiento se define como:

$$\begin{aligned} x_{ei} &= \cos \theta_i (x_{ri} - x_i) + \sin \theta_i (y_{ri} - y_i) \\ y_{ei} &= -\sin \theta_i (x_{ri} - x_i) + \cos \theta_i (y_{ri} - y_i) \\ \theta_{ei} &= \theta_{ri} - \theta_i \end{aligned} \quad (5)$$

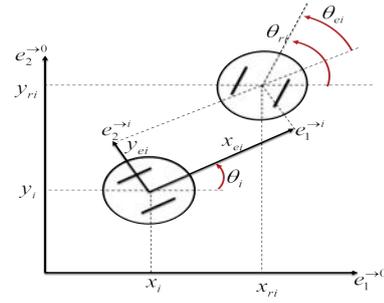


Figura 3. Posición actual y deseada de un robot diferencial (4).

Donde las coordenadas de referencia son x_{ri} , y_{ri} y θ_{ri} , mientras que las coordenadas reales son x_i , y_i y θ_i .

La dinámica de error está dada por:

$$\begin{aligned} \dot{x}_{ei} &= \omega_i y_{ei} - v_i + v_{ri} \cos \theta_{ei} \\ \dot{y}_{ei} &= -\omega_i x_{ei} + v_{ri} \sin \theta_{ei} \\ \dot{\theta}_{ei} &= \omega_{ri} - \omega_i \end{aligned} \quad (6)$$

Las trayectorias viables para el i -ésimo robot satisfacen las restricción no-holónoma, *i.e.* $\dot{x}_i \sin \theta_i - \dot{y}_i \cos \theta_i = 0$, por lo tanto para $\dot{x}_{ri} \neq 0$ y $\dot{y}_{ri} \neq 0$, las velocidades lineal y angular de referencia que se derivan de la trayectoria de referencia son:

$$v_{ri} = \sqrt{\dot{x}_{ri}^2 + \dot{y}_{ri}^2}; \quad \omega_{ri} = \dot{\theta}_{ri} = \frac{\dot{x}_{ri} \ddot{y}_{ri} - \ddot{x}_{ri} \dot{y}_{ri}}{v_{ri}^2}$$

El objetivo de la formación es el seguimiento cooperativo, esto es que cada robot siga su propia trayectoria de referencia de manera equilibrada, los errores de acoplamiento son definidos como:

$$\begin{pmatrix} x_{ei} - x_{ej} \\ y_{ei} - y_{ej} \end{pmatrix} \rightarrow \mathbf{0}, \quad \forall i, j \in (1, 2, \dots, n), \quad i \neq j \quad (7)$$

Se desea que cada robot siga su trayectoria de referencia, es decir, $x_{ei}, y_{ei}, \theta_{ei} \rightarrow 0$, y cumpliendo una forma definida, por ejemplo, cada trayectoria es diseñada para que los robots mantengan una distancia fija entre ellos emulando un eslabón virtual de un robot planar. Para lograr lo antes mencionado se utiliza el control propuesto en Morales *et al.* [2013], para las velocidades lineal y angular.

$$\begin{aligned} v_i &= v_{ri} \cos \theta_{ei} + \\ &C_{ix} \left[x_{ei} + \sum_{j=1, j \neq i}^n C_{ijx} (x_{ei} - x_{ej}) \right] \\ \omega_i &= \omega_{ri} + C_{i\theta} \theta_{ei} + v_{ri} \frac{\sin \theta_{ei}}{\theta_{ei}} \frac{K}{\alpha_i} C_{iy} \\ &\left[y_{ei} + \sum_{j=1, j \neq i}^n C_{ijy} (y_{ei} - y_{ej}) \right] \end{aligned}$$

Donde C_{ix} , C_{ijx} , C_{iy} , C_{ijy} , $C_{i\theta}$ y K son las ganancias del control y el término α_i delimita el efecto de los errores de seguimiento y acoplamiento, definiéndose como:

$$\alpha_i = \sqrt{K^2 + x_{ei}^2 + y_{ei}^2} + \beta_{ij} \quad (8)$$

$$\beta_{ij} = \sum_{j=1, j \neq i}^n (x_{ei} - x_{ej})^2 + \sum_{j=1, j \neq i}^n (y_{ei} - y_{ej})^2$$

Nota. Se puede apreciar que el control propuesto (ver (8)), existe un compromiso entre los errores de seguimiento x_{ei} , y_{ei} , θ_{ei} y los errores de formación $[(x_{ei} - x_{ej}), (y_{ei} - y_{ej})]$, lo que significa que a través de las ganancias C_{ix} , C_{iy} , $C_{i\theta}$ el objetivo de seguimiento se cumple, mientras que la formación es afectada por las ganancias C_{ijx} , C_{ix} y C_{ijy} , C_{ixy} , para cada x_i y y_i , respectivamente.

Los parámetros de diseño C_{ix} , C_{ix} y $C_{i\theta}$ son las ganancias del control responsables del seguimiento de las trayectorias de referencia para cada robot. Los términos C_{ijx} y C_{ijy} son las ganancias de acoplamiento, que afectan la coordinación de los robots en la formación. La ganancia K normaliza al término $\frac{K}{\alpha_i}$ cuando los errores de seguimiento y de acoplamiento son iguales a cero, por otra parte, limita al término $[y_i + \sum_{j=1, j \neq i}^n C_{ijy}(y_{ei} - y_{ej})]$ en el cálculo de la velocidad angular ω_i en (8). La estabilidad de este control fue demostrada en Morales *et al.* [2013].

5. TRAYECTORIAS CARTESIANAS DE REFERENCIA

Las trayectorias cartesianas de referencia se obtienen mediante una interpolación a nivel del espacio de configuración. Cada sucesión de configuraciones q_i obtenidas por el algoritmo Dijkstra se interpola con respecto al tiempo.

Dicha interpolación se hace durante un intervalo de tiempo $\Delta t = [0, t_{max}]$, donde t_{max} es el tiempo total que dura el experimento, lo que da como resultado una función continua a trozos $q_i(t)$ (ec. 4) que depende del tiempo y que representa la configuración que debe de adoptar el i -ésimo robot en cada instante de tiempo.

$$q_i(t) = \{\Delta t_k \quad m_k t + b_k\} \quad (9)$$

Con:

$$\Delta t_k = \frac{(k)t_{max}}{n} \leq t \leq \frac{(k+1)t_{max}}{n}$$

$$m_k = \frac{q_{ik} - q_{i(k-1)}}{\frac{(k+1)t_{max}}{n} - \frac{(k)t_{max}}{n}}$$

$$b_k = q_{ik} - m_k \frac{(k)t_{max}}{n}$$

Para $i = 0, 1, 2, 3$ y $k = 0, 1, \dots, n$, en donde i representa el número del robot y n representa en número total de configuraciones obtenidas por el algoritmo Dijkstra, incluyendo la configuración inicial y configuración final. Cada valor de la variable k representa un cambio tanto en la pendiente m_k como en el límite inferior y superior del intervalo de tiempo Δt_k .

En la figura 4 se puede apreciar de forma gráfica la interpolación lineal respecto al tiempo a nivel de espacio de configuración.

Finalmente, mediante cinemática directa de cada configuración $q_i(t)$, se obtiene la posición deseada en cada instante de tiempo para cada robot. La trayectoria de referencia para el i -ésimo robot se obtiene mediante:

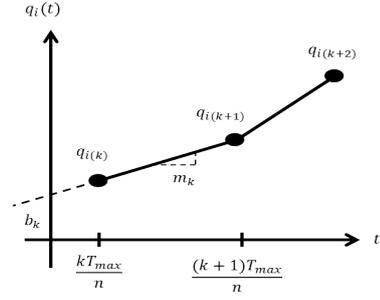


Figura 4. Interpolación lineal en el espacio de configuración.

$$x_{ri} = \sum_{j=1}^i l_j \cos \left(\sum_{k=1}^j q_k(t) \right) \quad y_{ri} = \sum_{j=1}^i l_j \sin \left(\sum_{k=1}^j q_k(t) \right) \quad (10)$$

Para $i = 1, 2, 3$.

En este caso, no es necesario obtener la trayectoria de referencia para el robot virtual $i = 0$ puesto que este robot es considerado la base del robot planar virtual, por lo cual nunca se traslada de su posición original, solamente gira sobre su propio eje, ver Figura 1.

Dichas trayectoria x_{ri} , y_{ri} son las coordenadas cartesianas requeridas por la ecuación (5) para obtener los errores cartesianos. Las derivadas \dot{x}_{ri} , \dot{y}_{ri} se obtienen de forma numérica y son necesarias para obtener la velocidad lineal, acorde a la ecuación (4), y el ángulo de orientación de referencia definido como:

$$\theta_{ri} = \arctan \left(\frac{\dot{y}_{ri}}{\dot{x}_{ri}} \right)$$

Por último, la velocidad angular ω_{ri} (ec. (4)) se obtiene mediante la derivada numérica de θ_{ri} .

La Figura 5 representa el diagrama general del funcionamiento de este trabajo. El primer recuadro punteado representa los algoritmos PRM-Dijkstra que entregan la trayectoria factible (TF) libre de colisión necesaria para que el grupo de robots vaya desde una configuración inicial a una configuración final, este proceso se efectúa fuera de línea. Con la trayectoria admisible encontrada se realiza la interpolación en el espacio de configuración. La interpolación de configuraciones para cada robot pasa al segundo recuadro punteado, que representa el proceso en línea, este proceso consiste en obtener las trayectorias de referencia para cada robot mediante la cinemática directa de las configuraciones interpoladas anteriormente. La derivada de esta trayectoria se calcula de forma numérica para posteriormente calcular la velocidad lineal y angular deseada que sirven como argumento de entrada al control. La posición de cada robot es obtenida mediante el sistema de visión en cada instante de tiempo y también es usada como argumento de entrada en el cálculo del control. Finalmente, la señal de control es enviada mediante comunicación bluetooth a los robots móviles.

6. EXPERIMENTOS Y RESULTADOS

Para validar el control propuesto se ejecutaron experimentos en CINVESTAV Unidad Saltillo. La plataforma experimental está conformada por tres robots móviles diferenciales de la marca iRobot Create®, una computadora de escritorio con Ubuntu 11.04, una cámara Infaimon

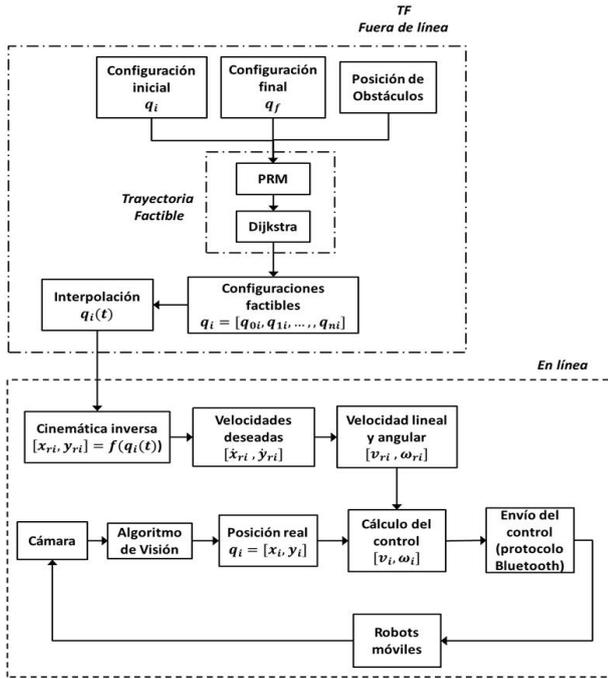


Figura 5. Diagrama de flujo para un grupo de robots móviles formando un robot planar virtual.

uEye® modelo UI-1220SE-M monocromática y módulos de comunicación Bluetooth®. La cámara está situada a 2 metros de altura y cubre un área de 2.3 metros de largo por 1.5 de ancho, ver Figura 6, los algoritmos fueron programados en C++ a través del editor Qt, se procesan las imágenes mediante un algoritmo de visión para obtener la posición y orientación de cada robot y calcular el control, por último se convierten las velocidades lineales y angulares a velocidades de rueda para cada robot y se envían mediante el protocolo de comunicación Bluetooth.

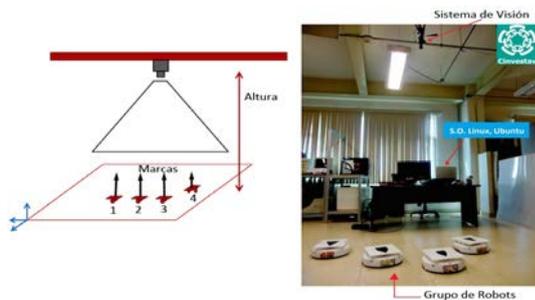


Figura 6. Plataforma Experimental.

6.1 Experimento

El experimento consistió en llevar al grupo de robots desde una configuración inicial a una configuración final sin colisionar con obstáculos estáticos. En la Tabla 3 se especifican los parámetros del experimento.

La Figura 7 hace referencia al experimento llevado a cabo, la línea color cian representa la configuración inicial, mientras que la línea amarilla representa la configuración final. La trayectoria deseada se obtuvo mediante la interpolación lineal de una sucesión de configuraciones factibles libres de colisión obtenidas por los algoritmos PRM-Dijkstra. La

Tabla 3. Parámetros del experimento.

Sucesión de Configuraciones						
	1	2	3	4	5	6
	(inicial)					(final)
q_1	-1.10	-1.30	-2.00	-0.90	0.10	0.45
q_2	1.10	1.30	1.90	2.20	0.50	0.00
q_3	0.00	0.10	1.50	-0.70	-0.40	-0.45
Eslabones Virtuales $l_1 = l_2 = l_3 = 0,5[m]$.						
Obstáculos						
	1		2			
Posición en X	1.250 [m]		0.220 [m]			
Posición en Y	0.005 [m]		0.550 [m]			
Largo l	0.500 [m]		0.700 [m]			
Ancho a	0.200 [m]		0.150 [m]			
Ganancias del Control						
$C_x=1.7, C_y=35, C_\theta=3.5, k=1, C_{ij}=1$ Para $i = j = 1, 2, 3$						

líneas roja, azul y verde representan las trayectorias seguidas por el primer, segundo y tercer robot respectivamente, los rectángulos morados representan los obstáculos y las líneas naranjas de los bordes de la figura representa los límites del espacio de trabajo, los cuales fueron modelados también como obstáculos.

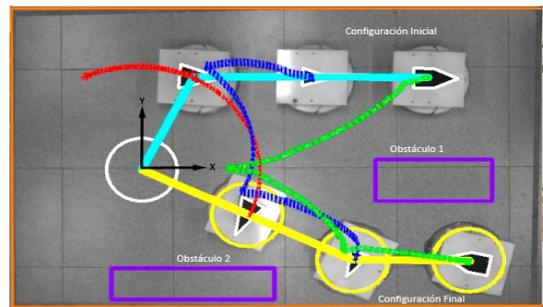


Figura 7. Trayectorias reales para robot en la formación.

Alguna de las configuraciones que adoptó la formación de los robots móviles en el experimento pueden verse en el fotograma de la Figura 8. Se observa que la estructura de la formación se mantiene, los robots llegan de la configuración inicial a la configuración final deseada sin colisionar con ningún obstáculo.

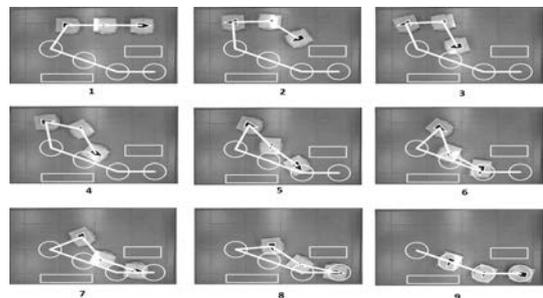


Figura 8. Algunas configuraciones adoptadas por la formación de los robots.

Durante el experimento, la longitud de los eslabones virtuales presentó solo algunas variaciones dentro de rangos

pequeños, ver Figura 9, lo que hace constar que la estructura de la formación de los robots se mantuvo estable.

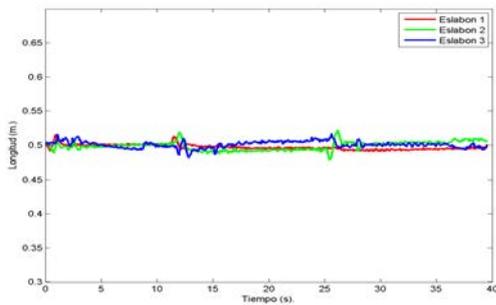


Figura 9. Relación de distancia entre robots (eslabones virtuales).

En la Figura 10 se puede apreciar los errores Cartesianos para los tres robots. Los sobresaltos observados en estas gráficas son originados por el cambio de orientación en la trayectoria, a pesar de ello, los errores en X y Y convergen a cero, lo que se traduce como el seguimiento efectivo de la trayectoria deseada.

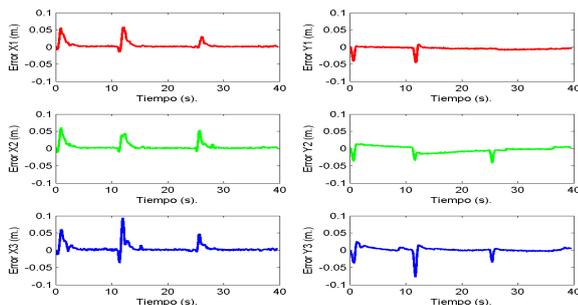


Figura 10. Errores Cartesianos para las trayectorias deseadas de cada robot móvil diferencial.

Con respecto a la velocidad lineal y angular calculadas por el control (Figura 11), se observa que presentan sobresaltos, relacionados con la magnitud de los sobresaltos de los errores Cartesianos en el instante de tiempo en que ocurren los cambios de trayectoria, esto como respuesta del control al hacer que los errores converjan a cero.

El video del experimento antes mencionado puede ser consultado en la siguiente dirección:

https://www.youtube.com/watch?feature=player_detailpage&v=EhjXBZIXLwA

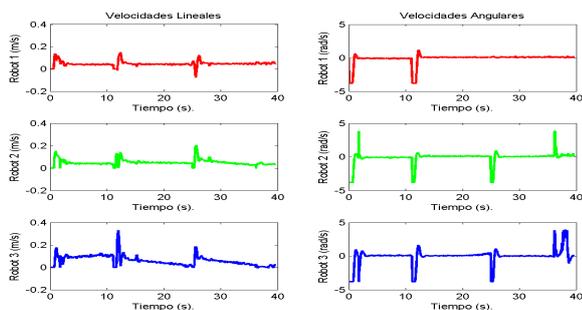


Figura 11. Velocidades lineal y angular para cada robot móvil diferencial.

7. CONCLUSIONES

En este artículo se presentó la navegación factible de un grupo de robots móviles diferenciales utilizando un enfoque de estructura virtual que permite mantener distancias entre los elementos del grupo y a la vez garantizar la evasión de obstáculos en el entorno.

El modelado de la formación de robots móviles como una estructura de un robot planar, donde cada unión de eslabones es representada por un robot móvil diferencial, permitió utilizar algoritmos probabilísticos como el PRM y Dijkstra para la planificación de trayectorias. La unión entre los vectores de configuraciones de salida se realizó mediante una interpolación lineal a nivel del espacio de configuración. Las configuraciones libres de colisión del robot planar virtual fueron convertidas a rutas cartesianas de referencia para cada robot móvil diferencial mediante cinemática directa. Por último el control propuesto demuestra que los robots cumplen con el seguimiento de la trayectoria de referencia por lo que se garantiza que no haya colisiones con obstáculos en el ambiente, así como choques entre los elementos del grupo.

Como trabajo futuro se tiene considerado que la cámara detecte la posición de los obstáculos de manera automática mediante el algoritmo de visión. En cuanto a la generación de la trayectoria admisible se planea utilizar otro método de interpolación para obtener una ruta más suave.

El control utilizado en este artículo se modificará para que el sistema sea robusto ante retardos en la comunicación, de tal manera que el seguimiento de la trayectoria de referencia obtenida por los algoritmos PRM y Dijkstra se garantice a pesar de la existencia de retardos en los sistemas de comunicación.

Finalmente se modelará la formación de los robots con otras estructuras virtuales, ingresando mas grados de libertad al robot planar virtual, por ejemplo articulaciones prismáticas que permitan la variación de longitud de los eslabones virtuales.

REFERENCIAS

- L.E. Kavraki, P. Svestka, J.C. Latombe, and M. Overmars (1996), Probabilistic Roadmaps for Path Planning in High-Dimensional Configuration Spaces. *IEEE Transactions on Robotics and Automation*, **12**(4), 566-580.
- D. Kostić, S. Adinandra, J. Caarls, N. van de Wouw, H. Nijmeijer (2010), Saturated Control of Time-Varying Formations and Trajectory Tracking for Unicycle Multi-agent Systems. *IEEE Conf. Dec. Control*, 4054-4059, Atlanta, GA.
- A. Sadowska, T. van den Broek, H. Huijberts, N. van de Wouw, D. Kostic, H. Nijmeijer (2011), A virtual structure approach to formation control of unicycle mobile robots using mutual coupling. *International Journal of Control*, **84**(11), 1886-1902.
- Y. Kanayama, Y. Kimura, F. Miyazaki, T. Noguchi (1990), A Stable Tracking Control Method for an Autonomous Mobile Robot. *IEEE International Conf. on Robotics and Automation*, 384-389.
- A. Morales, H. Nijmeijer, H. Gutiérrez (2013), A coordination control strategy for a group of unicycle robots. *Congreso anual Asociación de México de Control Automático*.