

## Conference Paper

# Algorithms for the Detection and Evasion of Obstacles on a Differential Mobile Platform for Interior Navigation

## Algoritmos para la Detección y Evasión de Obstáculos Sobre una Plataforma Móvil Diferencial para Navegación de Interiores

José Luis Tinajero, Pablo Lozada, Alonso Alvarez, and Narcisa Salazar

Escuela Superior Politécnica de Chimborazo, Código Postal 060155, RiobambaChimborazo, Ecuador

### Abstract

In this work, the algorithms necessary to execute displacements that allow indoor navigation for a differential mobile platform were designed, it was considered the programming of an Arduino Nano card connected to ultrasonic sensors that detect the distances of possible objects in the environment, and a Raspberry Pi3 card that is responsible for the activation and control of the platform's engines. A high level of efficiency of the System was achieved by evading obstacles and protecting the robot structure as much as possible

### Resumen

En este trabajo, se implementaron los algoritmos necesarios para ejecutar desplazamientos que permiten realizar navegación de interiores para una plataforma móvil diferencial, se consideró la programación de una tarjeta Arduino Nano conectada a sensores ultrasónicos los mismos que detectan las distancias de posibles objetos en el entorno, y una tarjeta Raspberry Pi3 que se encarga de la activación y control de los motores de la plataforma. Se logró un alto nivel de eficiencia del Sistema evadiendo los obstáculos y protegiendo al máximo la estructura del robot.

**Keywords:** algorithms, interior navigation, differential mobile platform, Arduino, Raspberry Pi3

**Palabras clave:** Algoritmo, Navegación de Interiores, Plataforma Móvil Diferencial, Arduino, Raspberry Pi 3.

Corresponding Author:

José Luis Tinajero

email:

joseluis.tinajero@espoch.edu.ec

Received: 4 December 2018

Accepted: 5 December 2018

Published: 27 December 2018

Publishing services provided by  
Knowledge E

© José Luis Tinajero et al. This article is distributed under the terms of the [Creative Commons](#)

[Attribution License](#), which permits unrestricted use and redistribution provided that the original author and source are credited.

Selection and Peer-review under the responsibility of the SIIPRIN-CITEGC Conference Committee.

### OPEN ACCESS

## 1. Introducción

Actualmente el desarrollo de robots completamente autónomos permite facilitar, simplificar y mejorar las condiciones de vida de sus usuarios en diferentes campos de aplicación. Para asegurar el funcionamiento e integridad de las estructuras, se los diseña de acuerdo a las necesidades de la aplicación para lo cual se instalan: cámaras, sensores y diversos actuadores[1].

Se destacan el trabajo denominado orientación visual bioinspirada en ambientes submarinos turbios que tiene por finalidad evitar obstáculos utilizando visión robótica monocular en escala de grises para entornos de agua turbia, considerando que las técnicas de evasión fueron diseñadas para ser computacionalmente de bajo costo [2], en la conferencia de Robótica y automatización presentada en el año 2010 se presentó el trabajo titulado Multi-Robots persecución y evasión sin mapas se desarrolla un algoritmo utilizando el seguimiento elemental de la pared y los comportamientos de descubrimiento de obstáculos[3], en el trabajo titulado un algoritmo practico de persecución-evasión: detección y seguimiento empleando una estrategia de detección de colisiones perezosas para resolver colisiones con obstáculos en el tiempo de ejecución[4], algoritmo de evasión de búsqueda en un ambiente desconocido propone un algoritmo distributivo de evasión de búsqueda para los múltiples agentes con rangos de detección limitados para perseguir a los múltiples evasores en un mapa de cuadrícula desconocida con obstáculos[5].

La presente investigación tiene como finalidad implementar los algoritmos de programación que permiten evadir cualquier tipo de obstáculos para realizar navegación por interiores aplicado a una plataforma móvil diferencial de modo que sea capaz de desplazarse de manera óptima y adecuada por cualquier entorno. Para cumplir con dicho objetivo, se empleó el microprocesador Raspberry Pi 3 conectado de manera inalámbrica con el software Matlab en el cual se indican los desplazamientos y giros realizados por la plataforma debido a las condiciones establecidas en la programación realizada en dicho software de manera adicional se emplea una tarjeta Arduino nano conectada a sensores ultrasónicos que indican la distancia a la cual se encuentra un determinado obstáculo.

## 2. Robótica

Se debe indicar que la robótica posee una historia bastante antigua basada en el desarrollo cultural de la humanidad. Desde siempre el humano ha buscado dispositivos

electromecánicos que sean capaces de sustituir los diversos comportamientos propios y característicos de los hombres en las diversas interacciones con el medio en el que habita, todo este avance tecnológico se lo ha generado por diversos aspectos en los cuales se puede destacar el filosófico, social, económico y científico [6].

El término robótica crea en nuestra mente dispositivos electromecánicos diseñados para la realización de rutinas repetitivas para la generación de trabajos productivos en áreas que pueden resultar de difícil acceso o peligrosas para los humanos, de forma adicional cabe indicar que son máquinas que imitan los movimientos y determinados casos el comportamiento de los seres humanos [7].

En la actualidad todo el desarrollo industrial y tecnológico se ha visto influenciado por dichos mecanismos, en sus inicios estas máquinas servían tan solo para magnificar la potencia muscular humana, debido al desarrollo tecnológico dichas máquinas son capaces de adquirir datos, analizarlos, interpretarlos y tomar las decisiones en base a condiciones y parámetros previamente establecidos por los programadores, de modo, que dichos dispositivos han sido capaces de sustituir a los humanos de forma completa en determinadas actividades [7].

## 2.1. Esquema General del Sistema Robot

La principal característica de funcionamiento de un robot se seccionada en diferentes áreas, las cuales trabajan de forma conjunta y en las cuales se destaca:

El sistema mecánico está conformado por los elementos que conforman la estructura de la máquina, es decir, un brazo articulado o un vehículo móvil, en el sector industrial la mayoría de los robots se encuentran sujetos a una base fija.[8]

Los sensores se encargan de dotar de información sobre la parte interna y externa del robot. Se diferencia dos tipos de sensores el primero de ellos son los internos que se encargan de indicar el estado de la estructura mecánica, giros, desplazamientos, fuerzas y pares, mientras que los sensores externos se encargan de proporcionar toda la información que rodea a la máquina.[9]

El sistema de control se encarga de realizar el procesamiento de la información obtenida de todos los sensores ubicados sobre la máquina[9].

Los actuadores responden a bucles o líneas de código que les permiten realizar acciones determinadas de acuerdo con las acciones tomadas por el sistema de control. Los actuadores en un robot pueden ser pinzas, motores, etc. [7] (Figura 1).

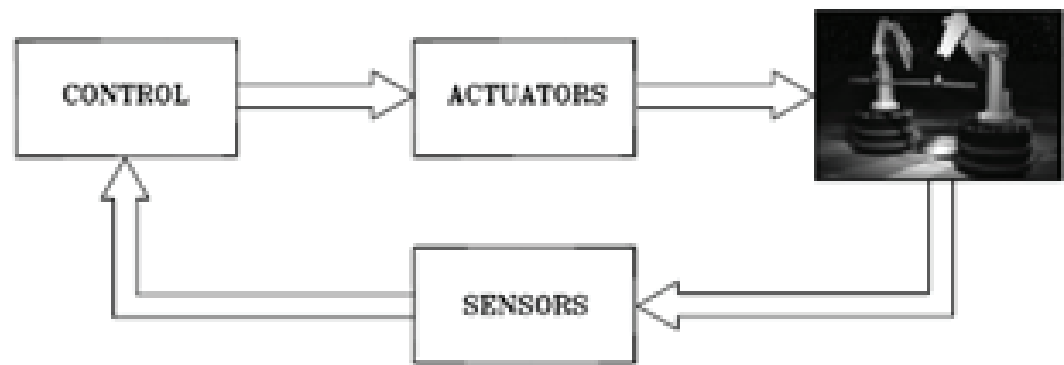


FIGURA 1: Esquema General del Sistema Robot.

## 2.2. Robots Móviles

Debido a la necesidad de extender el campo de la robótica, se desarrolló los robots móviles los mismos que conciben la idea de aumentar la autonomía de las máquinas reduciendo así la intervención de los humanos. La robótica móvil se enfoca en que el dispositivo tenga la suficiente inteligencia como para reaccionar y tomar decisiones acertadas basándose en los datos que puede tomar de su entorno, considerando que dicho medio no puede ser conocido completamente [6].

Una de las características más relevantes de los robots móviles es que estos poseen una base móvil que los permite desplazarse de forma libre por un ambiente conocido o desconocido, su principal aplicación es en la de servicio, es decir, aplicaciones donde necesita movimiento de un lugar a otro[1].

La base móvil denominada chasis es un cuerpo rígido que posee un sistema de locomoción el cual se encarga de realizar el movimiento de la máquina. Existen dos clases de robots móviles en las cuales se puede destacar:

- Los robots móviles con ruedas que están conformado por un chasis rígido y un sistema de ruedas que son las encargadas de proporcionar movimiento respecto al suelo [7].

## 2.3. Direccionamiento Diferencial

Su característica principal es que posee dos ruedas ubicadas en los costados del chasis, dichas ruedas se encargan de la tracción y direccionamiento del móvil. Para lograr el direccionamiento se debe variar las velocidades de las ruedas laterales. De forma adicional existe una o varias ruedas las cuales se emplean sobre la estructura para dar

soporte, pero no tienen función adicional. La aplicación más grande de esta configuración es para realizar desplazamientos por interiores [7] (Figura 2).

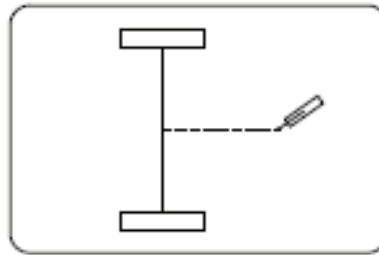


FIGURA 2: Diseño de una Plataforma Diferencial.

### 3. Controladores

#### 3.1. Controlador Raspberry PI 3

La tarjeta Raspberry Pi son pequeñas computadoras que fueron desarrolladas con la finalidad de promover los procesos de aprendizaje sobre conceptos básicos de informática en países en vías de desarrollo. La Raspberry Pi 3 es la tercera generación de tarjetas y reemplaza a la Raspberry Pi 2 modelo B, posee las siguientes características[10] (Figura 3).

- CPU: Quad Core 1.2GHZ Broadcom BCM2837 de 64bits
- Memoria RAM: 1GB
- Wireless LAN BCM43438 y Bluetooth Baja Energía (BLE) en la tarjeta
- 40-pin extendidos GPIO
- 4 USB 2 puertos
- Salida estéreo de poste y puerto de video compuesto
- Full size HDMI
- CSI puerto de camera para conectar a la Raspberry Pi Camera.
- DSI puerto para display para conectar el touchscreen display Raspberry Pi
- Puerto para MicroSD para cargar el sistema operativo y almacenamiento de datos
- Fuente de alimentación micro USB conmutada hasta 2,5 A

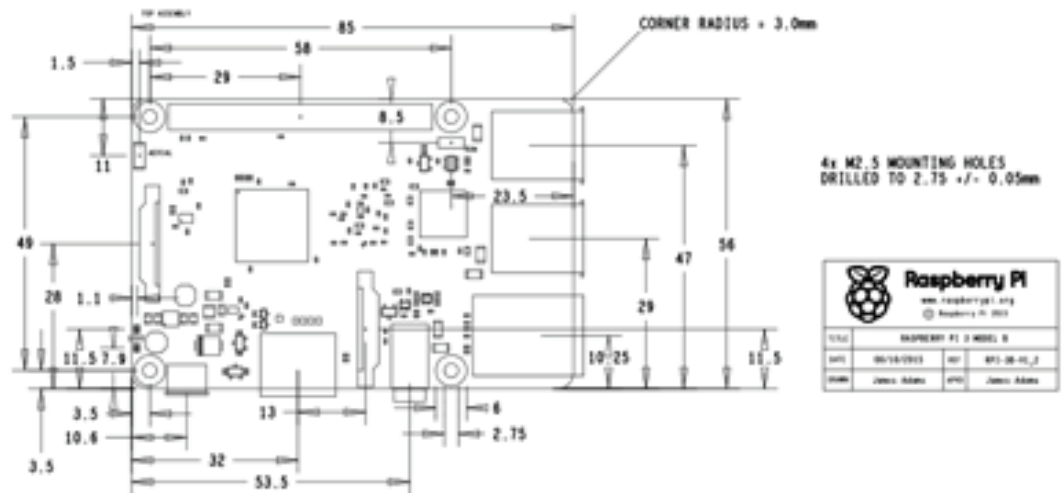


FIGURA 3: Diseño Placa Raspberry Pi.

### 3.2. Controlador de Motores

El dispositivo L293D es cuatro veces la mitad de los controladores H de alta corriente, está diseñado para proporcionar corrientes de accionamiento bidireccionales de hasta 600mA cuando trabaja en valores de voltaje de entre 4,5V a 36V. está diseñado para trabajar con cargas inductivas tales como relés, solenoides, motores paso a paso bipolares y motores de Corriente Continua, además de cargas que demandan alta corriente/alto voltaje en aplicaciones de suministro positivo. Cada salida es un circuito de accionamiento de tótem-pole completo, con un sumidero de transistor Darlington y una fuente pseud-Darlington, los controladores se habilitante en pares [11].

### 3.3. Arduino Nano

Es una placa bastante pequeña, completa y fácil de usar basado en el microcontrolador Atmega328, que ofrece la misma conectividad y especificaciones de la placa UNO en un formato más pequeño. Para programar un Arduino Nano se emplea el software de Arduino y al igual que el resto de placas puede funcionar en línea como fuera de línea[12] (Figura 4).

### 3.4. Sensor Ultrasónico

Es un sensor de distancia/proximidad cuya principal función es evitar obstáculos en diferentes proyectos de robótica, se emplean básicamente con las tarjetas Arduino y

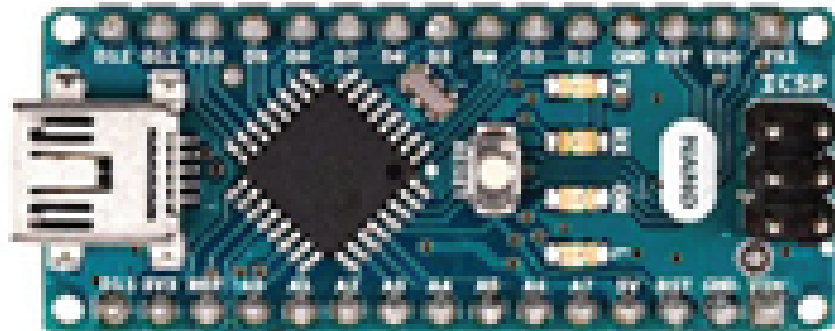


FIGURA 4: Tarjeta Arduino Nano.

suministran al proyecto información del entorno. Las aplicaciones de estos dispositivos son muy amplias y eficaces además del bajo costo económico que representa adquirirlos[13] (Figura 5).

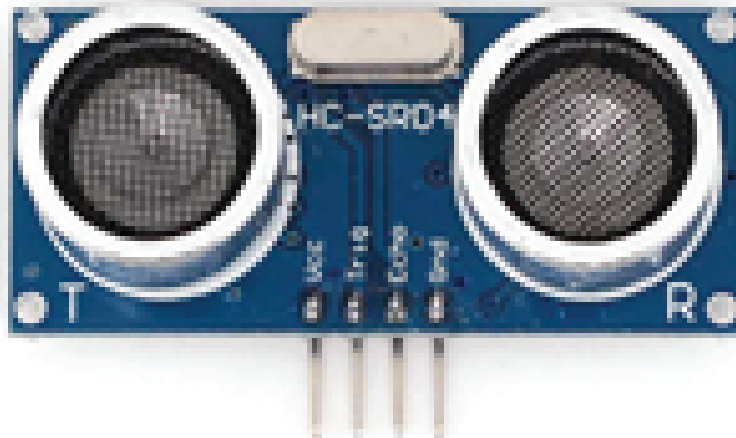


FIGURA 5: Sensor Ultrasónico.

## 4. Implementación En Arduino

### 4.1. Arduino

Es una plataforma de prototipos electrónica de código abierto compuesta por un microcontrolador, un lenguaje de programación y un IDE, es software y hardware libre, al ser código abierto se puede desarrollar una gran variedad de proyectos de electrónica con carácter multidisciplinario. El hardware Arduino es una placa que posee entradas digitales y analógicas por lo cual es ideal para una gran variedad de proyectos[12].

## 4.2. Delimitación del área del trabajo

Se realiza la delimitación del área de trabajo de la plataforma, para lo cual se consideró las dimensiones de la estructura y los rangos de funcionamientos de los sensores empleados. Los parámetros se los detalla en la Tabla 1.

TABLA 1: Parámetros de Delimitación del Área de Trabajo.

Parámetro	Valor(cm)
Distancia Máxima	300
Distancia Mínima	8
Distancia de Bloqueo	6

## 4.3. Diseño del Software en Arduino para detección de obstáculos

Se desarrolló un diagrama de flujo el que permite tomar la mejor decisión de desplazamiento considerando siempre preservar la estructura mecánica y elementos que componen al robot. Este diagrama de flujo sirve de base para la implementación del algoritmo de detección de objetos que, implementado en el microprocesador, permitirá controlar y evadir objetos delante de la plataforma móvil (Figura 6).

## 5. Implementación en Matlab

### 5.1. Matlab

Es una sofisticada herramienta computacional que tiene la finalidad resolver problemas matemáticos, destaca en cálculos numéricos relacionados a matrices y gráficas en los cuales se evidencien cálculos repetitivos y con gran cantidad de datos. Su aplicación no está limitada a un determinado enfoque específico[14].

### 5.2. Establecer conexión entre la Tarjeta y Matlab

Para establecer la conexión entre la tarjeta Raspberry Pi 3 y Matlab se crea un nuevo script y se ingresan las siguientes líneas de código (Figura 7):

```
clear all; clc;  
mypi = raspi;
```



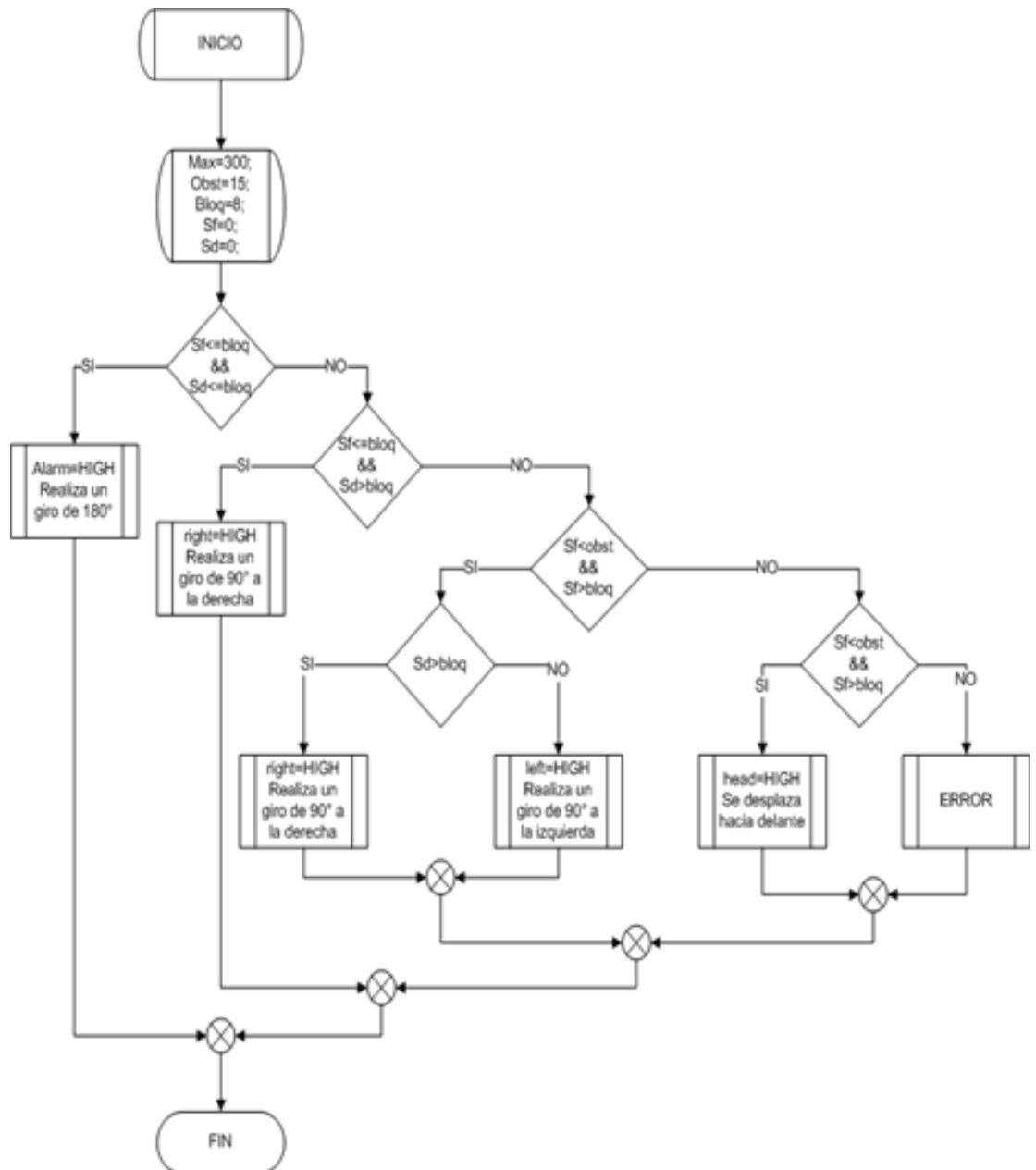


FIGURA 6: Diagrama de Flujo Arduino.

### 5.3. Accionamiento de motores

Previo al accionamiento de los motores se realiza la siguiente configuración en donde se declaran los pines de la tarjeta Raspberry que son empleados como pines de lectura y escritura respectivamente:



```
Command Window
New to MATLAB (see resources for Getting Started)

>> mypi=mypi

mypi =

  mypi with properties:
    DevAddress: 192.168.1.2
      Port: 10782
    BoardName: Raspberry Pi 3 Model B
    AvailableGPIOs: (1400)
    AvailableDigitalPins: [4, 5, 6, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 27]
    AvailableI2CChannels: (1 'CE0', 1 'CE1')
    AvailableI2CDevices: (1 '12C-1')
    AvailableWebcams: (1 'mal service 28.1 (platform=omni2888-v4l2)')
      I2CBusSpeed: 0

Supported peripherals
[?] [x] |
```

FIGURA 7: Conexión Entre Matlab y la Tarjeta Raspberry.

```
configurePin(mypi,13,'DigitalInput'); % activacion motores adelante
configurePin(mypi,5,'DigitalInput'); % activacion motores giro derecha
configurePin(mypi,23,'DigitalInput'); % activacion motores giro izquierda
configurePin(mypi,24,'DigitalInput'); % desactivacion de los motores por seguridad
configurePin(mypi,17,'DigitalInput'); % activacion ciclo repetitivo
configurePin(mypi,20,'DigitalOutput'); % rueda derecha
configurePin(mypi,21,'DigitalOutput'); % rueda derecha
configurePin(mypi,19,'DigitalOutput'); % rueda izquierda
configurePin(mypi,26,'DigitalOutput'); % rueda izquierda
configurePin(mypi,6,'DigitalOutput'); % señal desactivacion secuencial
configurePin(mypi,27,'DigitalOutput'); % indicador de obstaculo
```

Se considera las lecturas en los pines de la Raspberry, los mismo que son activados de acuerdo con la programación y decisión tomada por tarjeta Arduino Nano.

```
front=readDigitalPin(mypi,13);
right=readDigitalPin(mypi,5);
left=readDigitalPin(mypi,23);
top=readDigitalPin(mypi,24);
```

De acuerdo con la señal de activación presente en la tarjeta Raspberry se realiza el desplazamiento de la plataforma considerando cuatro condiciones importantes:

- FRONT: No existe un obstáculo delante de la plataforma móvil diferencial, por lo cual el robot debe desplazarse hacia delante.
- RIGHT: La plataforma presenta un obstáculo delante de ella, pero el sensor colocado a la derecha indica que el giro debe realizarse a ese lado.

- LEFT: Existen obstáculos en la parte frontal y derecha de la plataforma para lo cual se realiza un giro de 90° al lado izquierdo.
- TOP: La plataforma se encuentra en un callejón, para lo cual activa una alarma sonora para indicar que corre peligro la estructura y realiza un giro de 180°.

Se adjunta la respectiva configuración en uno de los casos:

```
if front
    % avance rueda derecha configuracion pin 20-0 pin 21-1
    writeDigitalPin(mypi,20,0)
    writeDigitalPin(mypi,21,1)
    % avance rueda izquierda configuracion pin 19-1 pin 26-0
    writeDigitalPin(mypi,19,1)
    writeDigitalPin(mypi,26,0)
    pause(.75)
    writeDigitalPin(mypi,20,0)
    writeDigitalPin(mypi,21,0)
    writeDigitalPin(mypi,19,0)
    writeDigitalPin(mypi,26,0)
    pause(0.5)
    writeDigitalPin(mypi,6,1)
    pause(.5)
    writeDigitalPin(mypi,6,0)
end
```

## 6. Pruebas

### 6.1. Detalles del experimento

Para la verificación del proyecto se plantea diversos escenarios los mismos que permite comprobar nivel de eficiencia de la programación para la evasión de obstáculos, las pruebas arrojan valores repetitivos por cada 5 pruebas realizadas se obtienen varios similares, las pruebas planteadas se las detalla en la Tabla 2:

### 6.2. Validación de datos

Se emplea el estadístico ANOVA pues permite comparar varias medidas realizadas de diversas situaciones o experimentos. Según los datos obtenidos basados a un nivel

TABLA 2: Resultados de las pruebas realizadas.

N° Prueba	Descripción	N° Total de Obstáculos	Obstáculos Evadidos	Eficiencia	Error
Prueba 1	Obstáculos: 2 Predominancia de Giro: derecha	10	9	90%	10%
Prueba 2	Obstáculos: 2 Predominancia de Giro: izquierda	10	9	90%	10%
Prueba 3	Obstáculos: 3 Predominancia de Giro: Híbrida	15	13	86,67%	13,33%
Prueba 4	Obstáculos: 3 Predominancia de Giro: Híbrida	15	13	86,67%	13,33%

de significancia del 5%, se encuentra que el punto crítico tiene un valor de 9,55. El valor calculado de Fisher es de 15,16 mediante la comparación entre estos dos valores se evidencia que el valor de Fisher se encuentra fuera de la región de aceptación, es decir, el punto crítico es menor que el valor calculado, por lo tanto, se considera que el funcionamiento del robot es óptimo (Figura 8).

Anova: Single Factor

SUMMARY					
Group	Count	Sum	Average	Variance	
Column 1	2	45	22,5	40,5	
Column 2	2	5	2,5	0,5	
Column 3	2	18,3	9,15	0,045	

ANOVA						
Source of Variation	SS	df	MS	F	P-value	Fcrit
Between Groups	414,9633333	2	207,4816667	15,16494	0,027004	9,332094
Within Groups	41,045	3	13,68166667			
Total	456,0083333	5				

FIGURA 8: Resultado del Estadístico ANOVA.

## 7. Conclusiones

Se diseñó el algoritmo que permite la correcta evasión de obstáculos considerando el mejor desplazamiento de la estructura móvil, siempre preservando la estructura, de manera que es capaz de desplazarse por cualquier entorno.

Matlab es un software basado en lenguaje C que permite implementar algoritmos en un lenguaje de alto nivel el mismo que puede ser ejecutado en dispositivos con

limitadas capacidades de procesamiento tal como microprocesadores, elementos utilizados en la solución del presente trabajo de investigación.

El sistema posee un excelente desempeño en la evasión de obstáculos, pero genera errores debido a condiciones externas a la programación y diseño del prototipo. Estos aspectos podrán ser corregidos en posteriores versiones de los algoritmos implementados.

## Referencias

- [1] Tinajero, J. L. " DISEÑO DE UN SISTEMA DE CONTROL VISUAL PARA NAVEGACIÓN DE INTERIORES USANDO FUSIÓN SENSORIAL PARA UNA PLATAFORMA MÓVIL DIFERENCIAL. (2018).
- [2] Song, Z., Schwartz, E. & Mohseni, K. Bioinspired Visual Guidance in Turbid Underwater Environment \*. 2-4 (2017).
- [3] Kolling, A. & Carpin, S. Multi-Robot Pursuit-Evasion without Maps. 3045-3051 (2010).
- [4] Aldahak, A. & Elnagar, A. A Practical Pursuit-Evasion Algorithm: Detection and Tracking. 10-14 (2007).
- [5] Chen, Y., Qi, H. & Liu, X. I. A. MAS-BASED PURSUIT-EVASION ALGORITHM UNDER UNKNOWN ENVIRONMENT. 18-21 (2005).
- [6] Siciliano, B., Sciavicco, L., Villani, L. & Oriolo, G. Robotics Modelling, Planning and Control. (2009). doi:10.1007/978-1-84628-642-1
- [7] Ollero, A. Robotica Manipuladores y robots moviles. Marcombo S.A (2001).
- [8] Holland, J. M. Designing Autonomous Mobile Robots. (2004).
- [9] Illah, R. Autonomous Mobile Robots.
- [10] Pi, R. Face Detection using MATLAB.
- [11] Texas Instruments Incorporated. L293x Quadruple Half-H Drivers. Texas Instruments Inc. 21 (2016).
- [12] Ramos Melgar, E. & Castro Diez, C. Arduino and Kinect Projects.
- [13] Jiang, N. Intelligent Stereo Camera Mobile Platform for Indoor Service Robot Research. Proc. 2016 IEEE 20th Int. Conf. Comput. Support. Coop. Work Desing 5-9 (2016).
- [14] Moore, H. Matlab para ingenieros. (2007).