VOLUMEN 21
XXVIII Verano De la Ciencia
ISSN 2395-9797
www.jóvenesenlaciencia.ugto.mx

Evasión de Obstáculos con un Robot Móvil Terrestre usando Aprendizaje por Refuerzo

M.A. Rojas Andrade, L. R. García Vázquez, E. Y. Aguilera Camacho, M.A. Escobar Carmona, A.S. Torres Villegas, J. P. Ramírez Paredes[1]
Departamento de Ingeniería Electrónica, Campus Irapuato-Salamanca
jpi.ramirez@ugto.mx[1]

Resumen

En este trabajo presentamos la aplicación del aprendizaje por refuerzo, a través del algoritmo *Q-learning,* para conseguir que un robot diferencial de cuatro ruedas evada obstáculos en un escenario. Utilizando un simulador de robots para la física del sistema, se ejecutaron múltiples episodios donde el robot intentó evadir obstáculos mientras se entrenaba el agente para que aprendiera a penalizar situaciones de colisión o posible colisión. Nuestros resultados muestran que, en unas horas, es posible conseguir un agente entrenado para evitar colisionar con obstáculos en múltiples escenarios. Comparamos nuestro agente con un vehículo de Braitenberg diseñado para evadir obstáculos, mostrando que el desempeño del agente rivaliza con el algoritmo clásico, pero no requiere que un experto diseñe la estrategia.

Palabras clave: Aprendizaje por refuerzo, robótica móvil, inteligencia artificial

Introducción

Dentro de la robótica, la evasión de obstáculos es la columna vertebral del control autónomo, ya que hace que el robot sea capaz de alcanzar destino sin colisión (Ye, Yung, & Wang, 2003). Esta problemática ha resultado en el estudio y evolución de múltiples técnicas de navegación a lo largo del tiempo, empezando por las ideas planteadas por Valentino Braitenberg en 1984, quién propuso vehículos teóricos sencillos que demostraron comportamientos inteligentes, como la evasión de obstáculos, a través de conexiones simples entre sensores y actuadores (Valverde Moreno, 2015). Los conceptos propuestos por Braitenberg sirvieron de referencia para la creación de modelos basados en sensores reactivos, lo que permitió a un agente moverse libremente de forma autónoma por un entorno (Lilienthal, A. & Duckett, T., 2004), evitando colisiones al utilizar un algoritmo simple mientras recibían información sensorial en tiempo real.

Otro enfoque de evasión de obstáculos es la implementación de *Q-learning*, el cual es un algoritmo de aprendizaje por refuerzo desarrollado por Christopher Watkins en 1989. Este enfoque permite entrenar a un robot diferencial para evitar obstáculos y navegar en entornos desconocidos. En este contexto, el robot actúa como un agente autónomo que toma decisiones en función de su estado actual y las recompensas recibidas por sus acciones (R. S. Sutton and A. G. Barto, 2018.). Este algoritmo *Q-learning* ha evolucionado a lo largo del tiempo a medida que se han realizado mejoras y variantes en el algoritmo original. Algunas de estas variantes son algoritmos de *Q-learning* modular que utilizan módulos fijos asignados por el usuario que combinan los resultados utilizando un enfoque simple conocido como el enfoque de la mayor masa. También se introdujo el Ant *Q-learning,* que combina el sistema de hormigas con el *Q-learning,* permitiendo que los agentes cooperen y compartan los valores Q. Además, se han propuesto enfoques jerárquicos de *Q-learning* para abordar el problema del aumento del espacio de estado-acción. Estos enfoques dividen el problema en niveles de abstracción, lo que permite una representación más compacta y eficiente del espacio de estado-acción (Jang, B., Kim, M., Harerimana, G., & Kim, J. W., 2019).

Existen otras variaciones de este algoritmo *Q-learning*, una de las más importantes, recientes y de mejores resultados es el algoritmo Deep Q-learning. El algoritmo Deep Q-learning aprovecha redes neuronales, tanto convencionales como convolucionales, para construir una aproximación de la función Q. Este resulta especialmente útil en problemas donde el agente tiene una gran cantidad de estados y acciones posibles (Fan, Wang, Z., Xie, Y., & Yang, Z, 2020).

Su planteamiento en el aprendizaje por refuerzo se basa en una red principal representada por los parámetros ø, se utilizan para estimar valores-Q del estado S y acciones actuales A. Utiliza una segunda red neuronal,

esta red neuronal es la red objetivo, parametrizada por ø', tiene la misma arquitectura que la principal pero se usara para aproximar valores-Q del siguiente estado S' y la siguiente acción A'. La red objetivo se congela después de varias iteraciones y después los parámetros de la red principal se copian a la red objetivo, transmitiendo así el aprendizaje de una a otra, haciendo que las estimaciones calculadas por la red objetivo sean más precisas (Van Hasselt, H., H., Guez, A., & Silver, D., 2016).

El aprendizaje por refuerzo con redes neuronales puede utilizarse para la evasión de obstáculos con robots móviles. En (Huang, Cao, & Guo, 2005) se usaron redes neuronales pera el entrenamiento del agente. El robot móvil estaba equipado con un sensor infrarrojo para la detección de obstáculos y generar el estado. El robot móvil podía realizar cinco acciones, avanzar hacia adelante, hacia la derecha, hacia la izquierda, rotación hacia la izquierda y rotación hacia la derecha. Este estado y acción era la entrada a la red neuronal la cual era una red de retro propagación donde almacenaba el valor Q. En la literatura también encontramos trabajos donde se controla la posición de un robot móvil utilizando aprendizaje por refuerzo profundo (Quiroga, Hermosilla, Farias, Fabregas, & Montenegro, 2022), o se emplean estos algoritmos para la navegación (Miranda, Neto, Freitas, & Mozelli, 2022).

En este trabajo mostramos que un agente que representa a un robot móvil terrestre, dotado de detectores de proximidad sencillos (detección binaria), puede ser entrenado para que automáticamente aprenda a desplazarse por un escenario mientras evade obstáculos. Mostramos una comparación entre este agente y un algoritmo inspirado en los vehículos de Braitenberg, que también pueden obtener una estrategia de evasión de obstáculos.

Materiales y Métodos

El aprendizaje por refuerzo es un paradigma dentro del aprendizaje de máquina, en el cual se representa un sistema como un agente inmerso en un entorno con el que interactúa. El agente puede ejecutar acciones, y realizar observaciones acerca de su propio estado. En el aprendizaje por refuerzo, el agente lleva a cabo acciones elegidas de acuerdo con algún método, de forma tal que al registrar los estados que alcanza tras dichas acciones pueda aprender cuáles conforman la mejor estrategia o política.

En este trabajo hacemos uso del algoritmo conocido como *Q-learning*. Este algoritmo se considera como "fuera de política" (*off-policy,* en inglés), ya que no requiere conocer la estrategia generada por el agente. En Q-learning, se aprovecha el hecho de que para aproximar la función Q, también llamada función acción-valor, no se requiere conocer pares de transiciones estado-acción como es el caso con el algoritmo SARSA. La función Q se aproxima utilizando diferencias temporales, de acuerdo con la siguiente ecuación:

$$Q(S_k, A_k) \leftarrow Q(S_k, A_k) + \alpha \left[R_{k+1} + \gamma \max_a Q(S_{k+1}, a) - Q(S_k, A_k) \right],$$

donde α, γ son hiperparámetros que determinan la razón de convergencia del algoritmo y se determinan de manera empírica. Durante el entrenamiento o aprendizaje de la tabla Q, es necesario alternar entre la exploración de pares estado-acción desconocidos, y la "explotación" o aprovechamiento de la función Q hasta el momento. Es por esto que se utilizan estrategias como la "ϵ-greedy", que en cada iteración escoge con probabilidad ϵ seleccionar una acción al azar, y con probabilidad $1 - \epsilon$ la acción que hasta ese momento resulte en el mejor valor Q. En nuestra implementación de *Q-learning* utilizamos el concepto de ϵ decreciente, de manera que la probabilidad de escoger acciones al azar disminuye de acuerdo con el número de episodios transcurridos. Esto es, la exploración cede el paso a la "explotación" según progresa el aprendizaje.

Para comparar el desempeño de *Q-learning* con otra técnica, también se implementó un algoritmo inspirado en los propuestos por V. Braitenberg. En este algoritmo, las velocidades de las ruedas del robot son directamente proporcionales a una suma ponderada de las lecturas de distancia de los sensores. El comportamiento resultante es que el robot gira en la dirección opuesta a la posición del obstáculo detectado. Este algoritmo es sencillo, mas no robusto, pues pueden aparecer situaciones donde el robot oscile o quede atorado al detectar obstáculos a distancias similares en dos sensores opuestos.

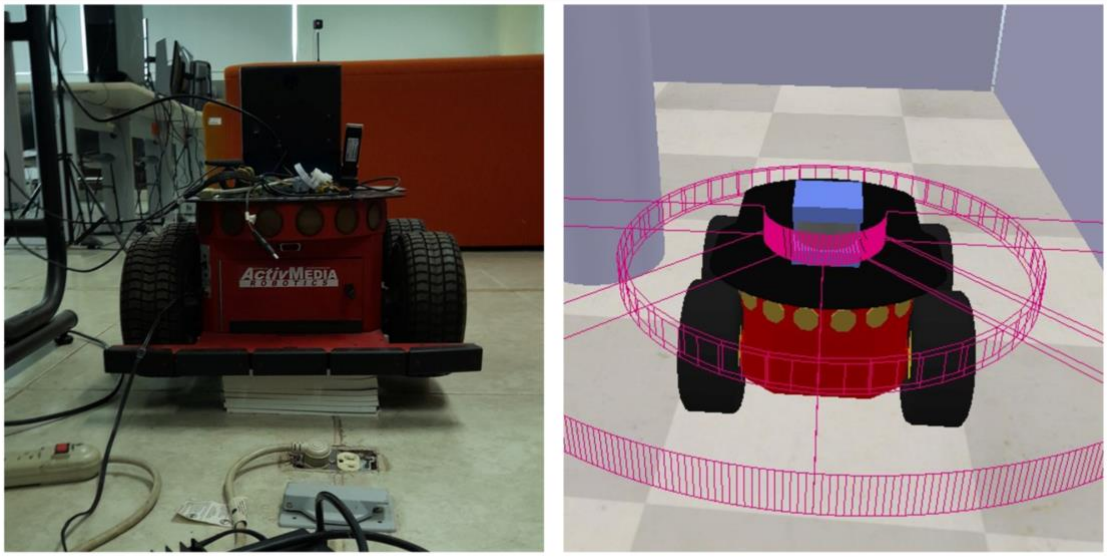

Figura 1: A la izquierda, el robot Pioneer P3-AT en el laboratorio. A la derecha, versión simulada del robot dentro del programa CoppeliaSim.

El robot utilizado en este trabajo es un Pioneer P3-AT, de la empresa Omron Adept. Es un robot terrestre con propulsión diferencial, con motores independientes para las llantas del lado izquierdo y el lado derecho. Posee un total de 4 llantas, utiliza una batería de ácido-plomo de 12V y está diseñado para ser una plataforma autónoma. Posee una computadora interna que ejecuta el sistema operativo Linux y la capa de software ROS (de *Robot Operating System*). El robot Pioneer P3-AT se encuentra en el Laboratorio de Visión, Robótica e Inteligencia Artificial del Campus Irapuato-Salamanca en la Universidad de Guanajuato, y se ilustra en la Figura 1

Para simular tanto el robot como su escenario para el proceso de aprendizaje por refuerzo, se utilizó el simulador CoppeliaSim en su edición para aplicaciones educativas. Este simulador no incluye un modelo del robot Pioneer P3-AT, pero permite la creación de nuevos modelos de manera simple. En este trabajo se implementó un nuevo modelo de CoppeliaSim para simular al robot Pioneer P3-AT.

Para crear este nuevo modelo se usaron mallas de cada una de sus piezas y fue armado de manera manual, las piezas constaban de llantas, tapas de llantas, chasis, y parte superior, se agregó adicionalmente en la parte superior el modelo de sensor “2D laser scanner” que viene en el simulador, se retiró el sensor de este para solo usar la pieza y se agregaron sensores adecuados de proximidad.

La estructura del robot consta del robot Pioneer P3-AT como cuerpo principal del modelo, de él se desprenden 2 articulaciones de revolución que se agregaron a cada lado, un total de 4, su función es hacer girar la llanta según sea necesario, a cada una de estas se le agrupo con una forma primitiva en forma cilíndrica cuya función es ser el área de interacción de la llanta y a estas misma se le agruparon las mallas de la llanta y la tapa o rin de la llanta igualmente agrupadas entres sí, estas últimas vendrían siendo la estructura visual, ya que las formas primitivas y las articulaciones están ocultas. Adicionalmente del cuerpo principal del modelo se desprende el cuerpo, el cuerpo es una agrupación de mallas, las cuales son el chasis y la parte superior o tapadera de este, de estos se desprende lo que es la forma primitiva del modelo del sensor y a su vez de este se desprende lo que vendría siendo las mallas del modelo, en estas viene agrupado lo que son los sensores de proximidad utilizados.

El robot Pioneer P3-AT posee un telémetro láser modelo LMS200 de la marca SICK, mostrado en la Figura 2, cuyas características se muestran en la Tabla 1.

Tabla 1: Características generales del telémetro láser LMS200.

Telémetro láser	Ángulo de abertura	Resolución angular	Resolución / Precisión típica de la medición	Rango típico	Rango de temperatura	Tensión de operación
LMS200	180°	0.25°; 0.5°; 1°	10 mm/±15 mm	10 m	0 a +50 °C	≤ 24 V DC, ± 15 %

Figura 2: Telémetro láser SICK LMS200.

Para reducir la complejidad de la estrategia de evasión de obstáculos, simplificamos el telémetro láser agrupando sus lecturas en 4 regiones, considerando únicamente como detecciones positivas aquellos objetos encontrados a menos de 0.5 m de distancia del sensor. De esta manera, podemos caracterizar el estado del agente utilizando las lecturas de estos 4 sensores binarios virtuales, dando como resultado 2^4 posibles estados. En la Figura 3 se muestra el esquema de numeración de los sensores simulados.

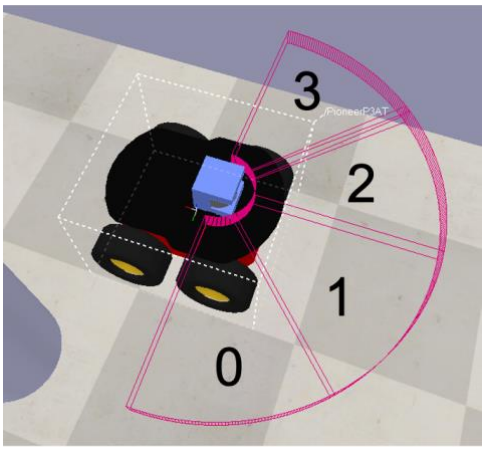

Figura 3: Enumeración de los sensores de proximidad de nuestro modelo del robot Pioneer P3-AT.

Aunque el rango de velocidades posibles para cada rueda del robot es amplio, y las velocidades permisibles son muchas, redujimos los movimientos permitidos al robot a solamente 6. Se describen las acciones de manera simplificada en la Tabla 2. Como se aprecia en la columna de recompensas, se incentivó la generación de movimientos de avance asignándoles una mayor recompensa que a los demás. El retroceder, aunque es una opción viable en muchas situaciones de colisión inminente, se penalizó con una recompensa negativa para indicar al algoritmo de aprendizaje que no es deseable en general este movimiento.

Tabla 2: Acciones del robot y sus recompensas

Número de acción	Descripción	Recompensa
0	Avanzar	20
1	Retroceder	-10
2	Giro abierto a la izquierda	5
3	Giro abierto a la derecha	5
4	Giro cerrado a la izquierda	5
5	Giro cerrado a la derecha	5

La simulación completa del robot en un ambiente con obstáculos se muestra en la Figura 4. Se agregaron muros perimetrales con los que el robot podría colisionar, así como obstáculos en forma de cilindros. Algunas esferas color verde indican los lugares donde el robot puede iniciar su tarea de evasión de obstáculos.

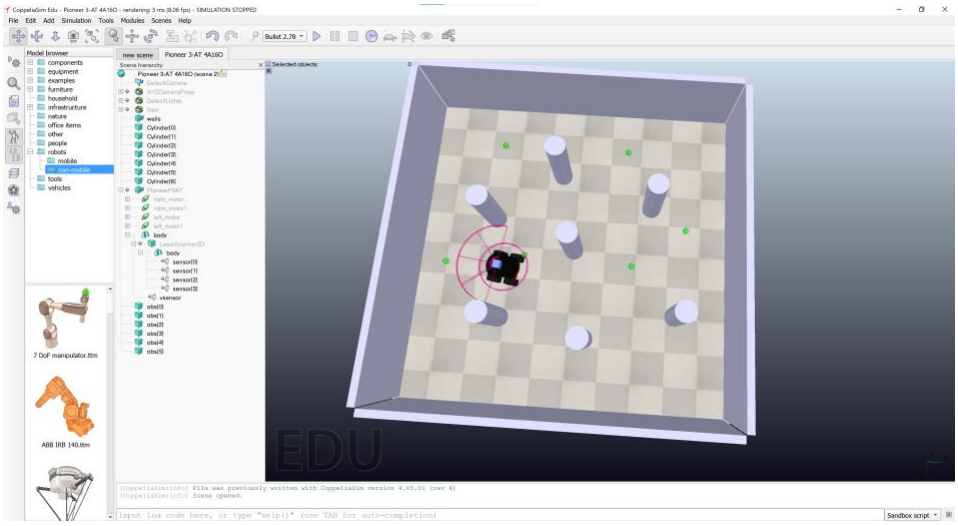

Figura 4: Escenario de simulación con el robot Pioneer P3-AT y obstáculos.

Resultados

Se llevó a cabo el entrenamiento del agente evasor de obstáculos usando 1000 episodios. Durante el entrenamiento se registró la recompensa acumulada de cada episodio. Cada episodio podía tener una duración máxima de 100 iteraciones; el episodio concluía al alcanzar ese número de iteraciones o al registrarse una colisión entre el robot y el ambiente, lo que ocurriera primero. Mostramos la recompensa acumulada en la Figura 5. Dado que durante los entrenamientos de estrategias de aprendizaje por refuerzo es normal una variación considerable en la recompensa obtenida entre entrenamientos individuales, en la **Error! Reference source not found.** de incluye también un promedio móvil de 10 muestras de la recompensa acumulada. Se puede verificar que durante los primeros 100 episodios existe un incremento importante de la recompensa. Si se continua el entrenamiento hasta los 1000 episodios se verifica que la recompensa promedio parece estabilizarse. La tabla Q tiene pocos estados y acciones asociadas, permitiendo así que el aprendizaje de la tarea se efectúe en pocas iteraciones.

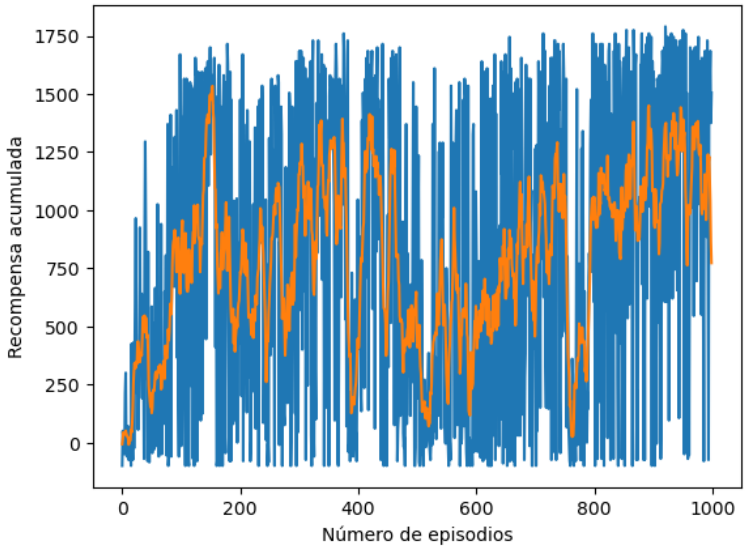

Figura 5: Recompensa acumulada durante el entrenamiento del aprendizaje por refuerzo con el algoritmo Q-learning. En azul: recompensa acumulada. En naranja: promedio móvil de 10 muestras de la recompensa acumulada.

Para verificar que el comportamiento del agente entrenado con *Q-learning* puede ser comparable o incluso mejor que otros algoritmos de evasión de obstáculos, se realizaron 100 simulaciones tanto del agente entrenado como del robot con navegación reactiva inspirada en vehículos de Braitenberg. Se registró el desplazamiento conseguido con respecto a la posición inicial en cada simulación. Los resultados se muestran en laTabla 3, utilizando el promedio y la desviación estándar de los desplazamientos. El agente entrenado con *Q-learning* logra, en general, desplazamientos mayores, por lo que, aunque ambas estrategias consiguen evitar las colisiones con obstáculos, el agente lo hace alejándose más del origen que el algoritmo tipo vehículo de Braitenberg.

Tabla 3: Métricas de desplazamiento de las dos técnicas de evasión de obstáculos bajo comparación.

Métricas de desplazamiento	Agente Q-learning	Algoritmo de vehículo Braitenberg
Desplazamiento promedio del robot (metros)	2.23	1.42
Desviación estándar del desplazamiento del robot (metros)	1.28	0.79

Conclusiones

En este trabajo se plantea y demuestra un método de aprendizaje por refuerzo, aplicado a la tarea de evasión de obstáculos por parte de un robot móvil terrestre. El modelo utilizado simplifica los sensores disponibles para la detección de los obstáculos, reduciéndolos a cuatro zonas de detección binarias, dando origen a 16 posibles estados de detección. También se simplificaron los movimientos posibles del robot, reduciéndolos a 6 acciones posibles de avance, retroceso y giros. El agente entrenado es capaz de avanzar mientras evade obstáculos tras mil episodios de entrenamiento. Esto es posible gracias al tamaño compacto de la función Q, debido al número de acciones y estados. Si se compara con otros problemas famosos en aprendizaje por refuerzo, esta función Q es notablemente pequeña.

El agente entrenado tiene un desempeño comparable o incluso mejor que el de una estrategia basada en vehículos de Braitenberg. Sin embargo, hay limitantes que no fueron resueltas o tratadas en este trabajo. Por ejemplo, la forma de los obstáculos es convexa y no se presentan muchas situaciones complejas donde exista ambigüedad sobre cuál es la mejor acción. Un ejemplo de esto es cuando el robot llega con un ángulo de 45 grados a una esquina formada por dos muros en ángulo recto: no es claro cuál acción es mejor para salir de esta situación, pues retroceder implica entrar en un ciclo de avance-retroceso, y girar hacia la izquierda o derecha puede no resultar en que el robot escape de esa zona del escenario.

Como trabajo futuro se plantea incrementar el número de estados posibles a través del uso de un modelo más complejo de sensor, incrementar el número de acciones admisibles y aplicar algoritmos de aprendizaje[i] por refuerzo mejor planteados para problemas de mayor complejidad, como DQN o DDQN.

Referencias

Fan, J., Wang, Z., Xie, Y., & Yang, Z. (2020). A theoretical analysis of deep Q-learning. In Learning for dynamics and control. 486-489.

Huang, B., Cao, G., & Guo, M. (2005). Reinforcement learning neural network to the problem of autonomous mobile robot obstacle avoidance. *International conference on machine learning and cybernetics* (págs. 85-89). Guangzhou, China: IEEE.

Jang, B., Kim, M., Harerimana, G., & Kim, J. W. (2019). Q-learning algorithms: A comprehensive classification and applications. 133653-133667.

Lilienthal, A., & Duckett, T. (2004). Experimental analysis of gas-sensitive Braitenberg vehicles. 817-834.

Miranda, V. R., Neto, A. A., Freitas, G. M., & Mozelli, L. A. (2022). On the Generalization of Deep Reinforcement Learning Methods in the Problem of Local Navigation. *Cornell University*, 3-13.

Quiroga, F., Hermosilla, G., Farias, G., Fabregas, E., & Montenegro, G. (2022). Position Control of a Mobile Robot through Deep Reinforcement Learning. *Applied Sciences*, 2-8.

R. S. Sutton and A. G. Barto, R. L. (2018.). *Reinforcement Learning: An Introduction.* MA, USA: MIT Press,.

Spano, S., Cardarilli, G. C., Di Nunzio, L, & Fazzolari, R. (2019). An efficient hardware implementation of reinforcement learning: The q-learning algorithm. 186340-186351.

Valverde Moreno, S. (2015). Robótica Inteligente: Implementación de sensores 3D para desenvolvimiento de robots móviles y vehículos autónomos. *Tecnológico de Costa Rica*, 19-79.

Van Hasselt, H., H., Guez, A., & Silver, D. (2016). Deep reinforcement learning with double q-learning.

Ye, C., Yung, N. H., & Wang, D. (2003). A fuzzy controller with supervised learning assisted reinforcement learning algorithm for obstacle avoidance. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 17-27.

Zohaib, M., Pasha, M, Riaz, R. A., Javaid, N., Ilahi, M., & Khan, R. D. (2013). Control strategies for mobile robot with obstacle avoidance. 1306.1144.

---

i