

## Implementación de un módulo de validación de acciones seguras para navegación de robots de servicio

### Safe actions module implementation for service robots navigation

PACHECO-FARFAN, L.†\*, RIOS-MARTINEZ, J, MORENO-SABIDO, M y SANDOVAL-CURMINA, V.

*Universidad Autónoma de Yucatán, Anillo Periférico Norte, T.C. 13615, Chuburná Hidalgo Inn, Mérida, Yucatán, México. Tecnológico Nacional de México/I.T. Mérida, Av. Tecnológico km. 4.5 S/N, C.P. 97118, Mérida, Yucatán, México*

ID 1<sup>er</sup> Autor: L. Pacheco-Farfan/ ORC ID: 0000-0003-4842-6396 , Researcher ID Thomson: T-3492-2018, arXiv ID Author: LPacheco, CVU CONACYT ID: 484010

ID 1<sup>er</sup> Coautor: J. Rios-Martinez/ ORC ID: 0000-0003-4208-0515, Researcher ID Thomson: H-1139-2018, arXiv ID Author: jorge.rios, CVU CONACYT ID: 280567

ID 2<sup>do</sup> Coautor: M. Moreno-Sabido/ ORC ID: 0000-0003-4690-7583, Researcher ID Thomson: T-3397-2018, arXiv ID Author: xacdc12, CVU CONACYT ID: 406193

ID 3<sup>er</sup> Coautor: V. Sandoval-Curmina/ ORC ID: 0000-0002-6511-7344, Researcher ID Thomson: T-3408-2018, arXiv ID Author: victor.sandoval, CVU CONACYT ID: 70654

Recibido 19 de Octubre, 2018; Aceptado 14 de Noviembre, 2018

#### Resumen

El diseño de sistemas de navegación autónomos para robots de servicio que trabajen en proximidad de humanos exige un adecuado balance entre tiempo de respuesta y garantía de seguridad para el robot y los humanos. En particular, se requiere un mecanismo para decidir qué estrategia de navegación es la mejor dependiendo de un criterio de seguridad. El presente artículo revisa enfoques de seguridad abordados en la literatura de robótica móvil y describe la implementación realizada del módulo PICS-DW, el cual asigna un valor de seguridad a las acciones de movimiento de un robot empleando un concepto llamado estados de colisión inevitable probabilísticos. Las acciones posibles de navegación se encuentran organizadas en una ventana dinámica y dependen de la configuración estructural del robot. El módulo se desarrolló en lenguaje C++ en el marco de trabajo del Middleware ROS (Robot Operating Systems) y la plataforma de simulación de robots MORSE. Los resultados muestran que el método implementado elige rutas de navegación que evitan zonas de riesgo alrededor de humanos simulados. Por la naturaleza modular del sistema y su licencia de código abierto, este prototipo puede ser usado por la comunidad para probar nuevas estrategias de navegación segura.

**Navegación autónoma, robótica de servicios, robótica computacional**

#### Abstract

The design of autonomous navigation systems for service robots which move near people requires an adequate balance between response time and safety for both humans and robots. Particularly, it is needed a mechanism to decide which strategy is the best according to a safety criterion. This article reviews different safety approaches appearing in the mobile robotics literature and describes the implementation of the software module PICS-DW which assigns a safety value to the possible navigation actions of a robot according to the concept of probabilistic inevitable collision states. The set of navigation actions are organized according to a Dynamic Window and are dependent on the structure of the robot. The module was developed using C++ language according to the framework of the Middleware ROS (Robot Operating System) and was tested on the robot simulation platform MORSE. The results show that the proposed method choose navigation paths which avoid risk zones around humans. Due to the modular nature of the system and its Open Source License the present prototype can be used by the community to test new safe navigation strategies.

**Autonomous navigation, service robots, computational robotics**

**Citación:** PACHECO-FARFAN, L., RIOS-MARTINEZ, J, MORENO-SABIDO, M y SANDOVAL-CURMINA, V.. Implementación de un módulo de validación de acciones seguras para navegación de robots de servicio. Revista de Ingeniería Innovativa 2018, 2-8: 11-21

\*Correspondencia al Autor (Correo Electrónico: gloria.mlg@hotmail.com)

† Investigador contribuyendo como primer autor.

## Introducción

En los últimos años, el interés en la creación y desarrollo de robots de servicio ha ido en aumento por su aplicabilidad en diversos campos como las tareas domésticas, la restauración, logística o vigilancia situación por la cual prometen mejorar la calidad de vida de las personas. Al mismo tiempo, la seguridad es un aspecto que debe tomarse en cuenta en el diseño de las interacciones con los humanos.

En robótica, la seguridad puede definirse de acuerdo a diversos aspectos ya que depende del propio robot, del ambiente (Chung et al, 2009) y del contexto con el cual interactúa. En la navegación autónoma de un robot la seguridad consiste en evitar que el robot cause colisiones con los diferentes elementos que se encuentran presentes en su ambiente (ISO131482, 2014), tanto aquellos que se mantienen fijos durante todo el proceso (como las paredes y los muebles), como aquellos que se encuentran en movimiento (como los humanos u otros robots).

Una de las estrategias que ha sido utilizada en la navegación de robots para evitar colisiones es el Enfoque de las Ventanas Dinámicas (Fox et al, 1995) o DWA por sus siglas en inglés. Este enfoque ha demostrado que es capaz de seleccionar movimientos o controles libres de colisión, sin embargo, durante la selección del control a realizar el algoritmo considera a todos los objetos detectados como estáticos. Lo que se suele hacer para contrarrestar esta deficiencia es ajustar los distintos parámetros del algoritmo para considerar ventanas de tiempo en el cual el cambio de posición de los elementos en movimiento sea tan pequeño como para ser negados y considerados elementos fijos.

Este trabajo busca mejorar la seguridad de la navegación de un robot cuando se usa el algoritmo DWA agregando una función de evaluación del riesgo de colisión aplicando el concepto de estados de colisión inevitable (ICS, por sus siglas en inglés) (Petti y Fraichard, 2005), el cual considera las posibles colisiones del robot con los objetos en movimiento tomando en cuenta las trayectorias futuras de los mismos. Al incluir esta función el algoritmo puede seleccionar controles más seguros.

El resto del documento se organiza de la siguiente manera, en la sección 2 se revisará el estado del arte, en la sección 3 se describirá el algoritmo DWA con las modificaciones realizadas en este trabajo, la sección 4 describirá el proceso realizado para la implementación del algoritmo, en la sección 5 se presentarán los resultados obtenidos, y finalmente, en la sección 6 se presentan las conclusiones y trabajo futuro.

## Antecedentes

La navegación en ambientes dinámicos requiere de la interacción de varios subprocesos tales como: encontrar una ruta desde la posición actual hasta la posición objetivo, seleccionar los movimientos a realizar o detectar los cambios en el ambiente. En la literatura se encuentran muchos trabajos que proponen enfoques para garantizar seguridad en la navegación de robots. Entendida como la ausencia de colisión, la seguridad debe equilibrarse con la eficiencia para seguir las trayectorias planeadas y alcanzar el objetivo por parte del robot. Como ejemplo de tales trabajos está el presentado por (Alami, Krishna y Simeon, 2007) donde se define un perfil de velocidad con base en la máxima velocidad que podría alcanzar un robot sin colisionar con los objetos de su entorno. (Aoude, Luders, Joseph, Roy, y How, 2013) combinan Procesos Gaussianos en las trayectorias de los objetos con algoritmos RRT (LaValle, 1998) para garantizar la ausencia de colisiones durante la navegación, mientras que (Bouraine, Fraichard y Salhi, 2012) aplican en la navegación de robots el concepto de seguridad en movimientos pasivos, es decir, si hay colisión el robot deberá estar en reposo. Un trabajo muy reciente que guarda bastante similitud con el que se plantea en este paper es el presentado por (Lorente, Owen, y Montano, 2018) en el cual se aplica un modelo llamado Dynamic Velocity Obstacles para describir el movimiento de los objetos, y con base en ello decidir cuál estrategia de las definidas en su método es la más segura.

A diferencia de los trabajos anteriores, esta propuesta se enfoca en evaluar la seguridad de los controles en etapas muy cercanas de la ejecución de la trayectoria seleccionada, porque el interés principal de la presente propuesta está en aquellos movimientos que le robot realiza muy cerca de los humanos.

Además, en el módulo desarrollado se utiliza un par de modelos para el movimiento de los humanos que dan sentido al espacio alrededor de ellos, por ejemplo, el modelo PS (Personal Space) da un mayor riesgo a la zona situada al frente del humano que a la situada en la posterior a diferencia de los otros métodos que tienden a considerar a los humanos como círculos con igual riesgo en cualquier parte.

De acuerdo a lo expuesto anteriormente, se decidió priorizar el tema de selección de controles seguros más que el de selección de trayectorias seguras a largo plazo. En ese sentido, se detectaron tres estrategias de seguridad que se enfocan en la selección de los controles: el Enfoque de Diagramas de Proximidad (NDA o simplemente ND por sus siglas en inglés) (Minguez y Montano, 2004), el Enfoque de Ventanas Dinámicas (DWA o DW por sus siglas en inglés) y el Enfoque de la Velocidad del Obstáculo (VOA por sus siglas en inglés) (Fiorini y Shiller, 1998).

Para esta propuesta se eligió el enfoque de Ventanas Dinámicas o DWA (Fox et al, 1995) el cual funciona creando un conjunto de velocidades alcanzables y admisibles a partir de la velocidad actual del robot. Una velocidad es considerada alcanzable si ésta puede ser alcanzada por el robot al aplicar una aceleración o desaceleración constante, durante un período de tiempo definido, a su velocidad actual y es considerada admisible si ésta le permite al robot detenerse antes de que ocurra una colisión con algún obstáculo. Este conjunto de velocidades es lo que se denomina ventana dinámica; posteriormente, se realiza para cada una de las velocidades alcanzables y admisibles una simulación para determinar cómo va cambiando el estado del robot al aplicar esa velocidad, y así evaluarla tomando en cuenta distintos parámetros, como la proximidad entre el robot y su destino, la proximidad entre el robot y los obstáculos, la propia velocidad que alcanzaría el robot después de aplicar la velocidad, y la proximidad entre el robot y el plan global (el plan global suele ser proporcionado por algún planeador global).

Una vez evaluadas todas las velocidades admisibles se elige como próximo control a ejecutar aquella velocidad que maximice los parámetros antes mencionados. Además, como se requería una forma de catalogar los controles con respecto a su seguridad, se incluyó el concepto de estados de colisión inevitable (ICS), es decir, estados desde los cuales, independientemente del control que se elija, el robot colisionará. Los ICS tienen una validez más teórica que práctica ya que calcularlos puede ser muy costoso computacionalmente. En (Bautin et al, 2010) se propone el concepto de ICS probabilístico que se define como la probabilidad de que un estado sea un ICS, para poder determinar este valor de probabilidad se requiere de un modelo probabilístico que le permita aproximar los movimientos futuros de los obstáculos dinámicos. El concepto de los PICS junto con el DWA son los dos conceptos principales implementados en este trabajo.

### Incorporando los PICS al DWA

Al incorporar el concepto de los PICS al DWA se evalúa cada uno de los controles que conforman la ventana creada con el DWA y se obtiene así el riesgo de colisión que posee cada uno de ellos. En este punto ya es posible seleccionar el control que mejor siga el plan global asignado al robot con el menor riesgo posible.

El proceso que se sigue se muestra en el diagrama de la Figura 1, donde se identifican tres etapas: creación de la ventana dinámica, evaluación de la ventana dinámica y selección del mejor control; a continuación, se describe cada una de esas etapas.

Dependiendo de sus capacidades y restricciones físicas y dinámicas, cada robot posee un espacio de velocidades que puede alcanzar, en (Fox et al, 1995) cada una de estas velocidades están representadas por curvas que se derivan de un par  $(v, \omega)$  que caracterizan las trayectorias que seguiría el robot al moverse con dicha velocidad en un determinado lapso de tiempo (aquí  $v$  es el componente de traslación de la velocidad o velocidad traslacional y  $\omega$  es el componente rotacional de la velocidad o velocidad rotacional).

Esto deriva en un espacio de búsqueda de velocidades bidimensional que por lo general es un conjunto de gran tamaño, que puede llegar a ser infinito; esto hace que la tarea de determinar cuál es la mejor velocidad para ejecutar sea bastante tardada. Para evitar eso, en la primera etapa de creación de la ventana dinámica, se reduce el espacio de búsqueda utilizando únicamente aquellas velocidades que son alcanzables por el robot a partir de su velocidad actual, bajo las restricciones dinámicas. Cabe aclarar que tales restricciones varían grandemente si se trata de un robot diferencial, un robot con patas o un robot humanoide. En el presente trabajo se modeló al robot como un robot diferencial.

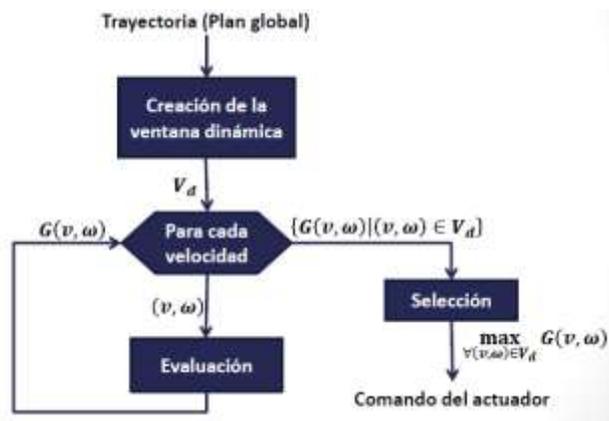


Figura 1 Diagrama de ejecución del algoritmo DWA  
Fuente: Elaboración Propia

El conjunto de velocidades que pueden ser alcanzadas son aquellas que resultan de aplicarles las aceleraciones y desaceleraciones  $\dot{v}$  y  $\dot{\omega}$  a los componentes rotacional y traslacional de la velocidad actual  $(v_a, \omega_a)$  como se muestra en la siguiente ecuación:

$$V_d = \left\{ (v, \omega) \mid \begin{array}{l} v \in [v_a - \dot{v} \cdot t, v_a + \dot{v} \cdot t], \\ \omega \in [\omega_a - \dot{\omega} \cdot t, \omega_a + \dot{\omega} \cdot t] \end{array} \right\} \quad (1)$$

El conjunto resultante  $V_d$  se llamará de ahora en adelante la “Ventana Dinámica” de donde se tomarán los controles posibles de movimiento para el robot.

El propósito de la etapa de evaluación es asignar una puntuación a cada una de las velocidades que permita posteriormente seleccionar la mejor opción. Para asignar esta calificación en (Fox et al, 1995) utilizan una función objetivo  $G(v, \omega)$  que se encuentra definida de la siguiente manera:

$$G(v, \omega) = \sigma(\alpha \cdot heading(v, \omega) + \beta \cdot dist(v, \omega) + \gamma \cdot velocity(v, \omega)) \quad (2)$$

Donde  $heading(v, \omega)$ ,  $dist(v, \omega)$  y  $velocity(v, \omega)$  son a su vez funciones que evalúan cada velocidad considerando únicamente un criterio;  $\alpha$ ,  $\beta$  y  $\gamma$  son constantes que ponderan el peso que se le otorga a cada criterio, y que por lo tanto ajustan las prioridades con las que se tomarán las decisiones en el proceso de la evasión de obstáculos. Por último,  $\sigma$  es una constante utilizada para normalizar los resultados de las evaluaciones. Los valores de tales constantes se tomaron de acuerdo a los estándares que ROS define en el modelo diferencial para su paquete de navegación.

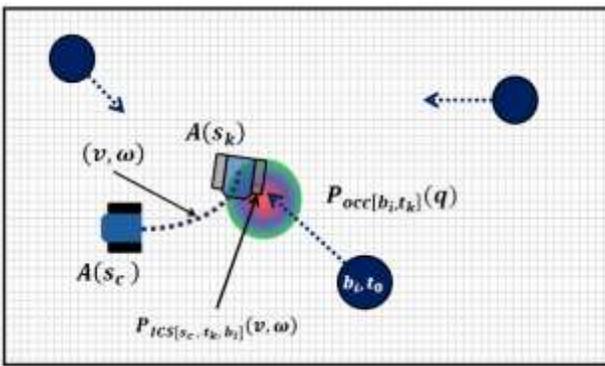
El criterio usado para determinar cuál es la mejor opción es elegir la velocidad que maximiza el valor de la función objetivo  $G(v, \omega)$ .

### Estados de Colisión Inevitable Probabilísticos (PICS)

Como se mencionó en la sección 2, el concepto de los PICS usa un modelo de ocupación de los objetos para estimar qué celdas estarían ocupadas en un futuro cercano.

En (Bautin et al, 2010) se definen procesos estocásticos independientes para cada uno de los parámetros de configuración de cada uno de los objetos  $\mathcal{B}_i$  en el conjunto de objetos  $\mathcal{B}$ , es decir, se usan variables aleatorias dependientes del tiempo para las coordenadas del punto de referencia de cada objeto  $\mathcal{B}_i$  y para su respectiva orientación. Así, a cada una de las variables aleatorias que corresponden a las coordenadas se les asocia una función de densidad la cual se emplea para obtener la probabilidad de que un punto  $(x_w, y_w)$  en el espacio de trabajo  $\mathcal{W}$  se encuentre ocupado por  $\mathcal{B}_i$  en el instante de tiempo  $t$ .

El proceso usado para encontrar la probabilidad de colisión con un objeto  $b_i$ , en un instante de tiempo  $t_k$  al realizar un control  $(v, \omega)$  se encuentra ilustrado en la Figura 2. Para calcular la probabilidad de colisión del objeto  $b_i$  con el robot  $A$  se estima la posible posición futura en ambos para el tiempo  $t_k$ . En particular se calcula la probabilidad de que, al elegir  $(v, \omega)$  de la ventana dinámica, el robot llegue a un estado de colisión inevitable.



**Figura 2** Cálculo de la probabilidad de ICS

Fuente: Elaboración Propia

Al utilizar un modelo probabilístico no es posible determinar si ocurrirá una colisión en el futuro de forma infalible. Debido a esto se asigna a cada estado una medida de probabilidad que indique que tan probable es que se trate de un ICS. Un estado  $s$  tendrá más probabilidad de ser ICS si todas las trayectorias de control en el espacio de control  $\tilde{U}$  tienen una probabilidad de colisión muy alta ( $\approx 1$ ) cuando parten de ese estado.

En este trabajo se modificó el algoritmo Forward PICS-Check (Bautin et al, 2010) para determinar la probabilidad de que un estado sea un ICS. El enfoque usado consiste en evaluar cada una de las velocidades en la ventana dinámica con los PICS para descartar a aquellas cuya trayectoria generada atraviesan celdas o estados con una probabilidad PICS mayor a un umbral definido; dado que las trayectorias generadas por cada velocidad son representadas por una secuencia de celdas o estados, se evalúa cada una de estas usando los PICS. Se modificó el algoritmo original de tal manera que en vez de evaluar un subconjunto del conjunto de trayectorias de control, se evalúen las velocidades de la ventana dinámica  $V_d$  del DWA, de tal forma que los conjuntos de probabilidades se realicen para un sólo control durante los instantes de tiempo que éste dura. Además, se modificó la función objetivo del DWA como se muestra en la ecuación 3 de manera que incorpore dicha probabilidad como una función de costo para las velocidades candidatas  $(v, \omega)$ ; de esta manera se espera que el algoritmo seleccione el control que se aproxime mejor a la trayectoria establecida o al objetivo de navegación sin comprometer la seguridad del robot y de los objetos  $\mathcal{B}_i$  presentes en el espacio de trabajo.

$$G(v, \omega) = \sigma \left( \alpha \cdot \text{heading}(v, \omega) + \beta \cdot \text{dist}(v, \omega) + \gamma \cdot \text{velocity}(v, \omega) + \delta \cdot \left( 1 - P_{ICS[v, \omega]}(s) \right) \right) \quad (3)$$

El valor de la variable  $\delta$  en la ecuación 3 permite ajustar la prioridad que le dará el DWA a la probabilidad PICS.

## Implementación

La implementación se realizó con la integración de diferentes bibliotecas y paquetes computacionales disponibles para trabajar en robótica, como son el paquete de navegación de ROS, el ambiente de simulación MORSE y la biblioteca de planeación de movimientos OMPL para aprovechar su implementación del algoritmo RRT\* (Karaman y Frazzoli, 2011). Se usó un modelo de ocupación para el robot que toma como base el modelo de transición de los robots con ruedas diferenciales.

Se realizó una extensión al módulo DWA que ya tiene ROS modificando tanto la clase "mobile\_robot\_kinematic\_model" como la clase "probabilistic\_occupation\_model" y añadiendo una clase "pics". Las dos primeras son abstracciones que modelan de manera general los modelos de predicción del futuro del robot y de la ocupación de las celdas por los obstáculos dinámicos respectivamente; estas funcionan como plantillas para trabajar con distintos modelos de ocupación y de robots.

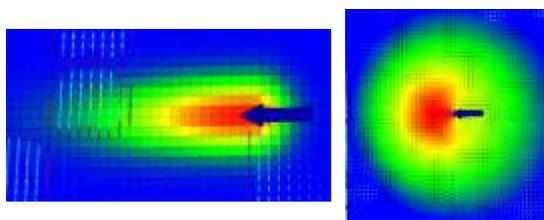
Los modelos de predicción de la ocupación futura de los obstáculos indican la probabilidad de que una celda esté ocupada por el obstáculo  $b_i$  en el instante de tiempo  $t$ . Para utilizar los PICS se implementaron 2 modelos (Rios-Martinez, 2013) que coinciden con el "Espacio personal (PS)" y con el "Espacio de Procesamiento de Información (IPS)" de los humanos. Tanto la implementación del modelo PS, como el del IPS, constan de dos pasos principales:

- Predecir la posición del centro de los obstáculos (o humanos) en el instante de tiempo  $t$ .

- Construir el área de ocupación a partir de la posición del centro del obstáculo usando el modelo especificado.

Para realizar el primer paso en ambos modelos, se asume que los humanos no cambiarán de velocidad ni de dirección durante los siguientes  $t$  segundos.

La Figura 3 muestra el área de ocupación obtenida para cada modelo. La flecha azul representa la posición y la orientación del humano para el cuál se realizó la predicción. Las probabilidades más altas se muestran en color rojo y que corresponden a probabilidades arriba de 0.75. El color amarillo corresponde a probabilidades entre 0.5 y 0.75, el verde a probabilidades entre 0.25 y 0.5, y los azules a probabilidades inferiores a 0.25.



**Figura 3** Áreas de ocupación. Implementación del modelo PS a la izquierda y el modelo IPS a la derecha  
Fuente: *Elaboración Propia*

El espacio personal representa un espacio alrededor de un humano en el cual se siente incomodidad cuando otra persona se acerca, generalmente es más amplio adelante que a los lados. El modelo IPS se basa en la formulación del efecto Doppler el cual establece que la percepción en la frecuencia de un sonido varía con el movimiento de la fuente y el observador. Ambos modelos están pensados para ser utilizados en conjunto con un sistema de seguimiento, el cual será el encargado de proporcionarle al modelo la posición actual de la persona con respecto al marco de referencia de  $\mathcal{W}$ , así como su velocidad traslacional y rotacional que es la información que el modelo requiere para poder estimar la posición y orientación que tendrá esa persona dentro de  $t$  segundos y, a partir de allí, encontrar el valor de probabilidad, según el modelo usado, para la celda que se requiera evaluar, el valor obtenido será utilizado como la probabilidad de ocupación de dicha celda.

Las pruebas realizadas en este trabajo se hicieron por medio del simulador MORSE (siglas en inglés de Modular OpenRobots Simulation Engine) el cual es un simulador genérico para propósitos académicos desarrollado inicialmente en el laboratorio francés “LAAS-CNRS”. Se utilizó el ambiente de simulación del laboratorio “LAAS-CNRS” que es proporcionado en el paquete de instalación de MORSE, el cual incluye un modelo de robot PR2 y se le agregaron dos modelos de humanos como se aprecia en la Figura 4. El robot PR2 incorpora sensores láser y de odometría además posee actuadores para el control de sus partes móviles y de sus ruedas. Tales sensores y actuadores simulados proporcionan información al módulo de ROS conocido como Navigation Stack el cual se encarga de realizar la localización en el mapa, determinar el plan global para llevar al robot a su posición objetivo y por último ejecutar los controles seleccionados. El módulo desarrollado en este trabajo se comunica con el Navigation Stack. En esta simulación se ha optado por controlar a los humanos manualmente con el teclado a los cuales se le ha asignado un sensor que permite saber su posición.

El modelo del PR2 proporcionado en MORSE ya está configurado para su uso con ROS. Esto no ocurre con los humanos del simulador, en la aplicación, los datos publicados por el sensor asociado a cada uno de los humanos, es considerado como parte del sistema de seguimiento. Bajo este supuesto se creó un componente en ROS que toma las lecturas de los sensores de los humanos en MORSE, convierte esta información a los tipos de datos definidos en ROS y las concentra en un solo arreglo para luego proporcionárselo al módulo que implementa los algoritmos de PICS.



**Figura 4** Modelo 3D del laboratorio. Se observa el escenario con un robot PR2, dos humanos y diferentes muebles

Fuente: *Elaboración Propia*

## Resultados Obtenidos

Las pruebas realizadas en este trabajo se enfocan en mostrar el efecto que se tiene en la toma de decisiones del planeador local cuando se incluye la modificación hecha de PICS.

Se utilizó la implementación del PICS-DWA, los modelos de ocupación para los obstáculos descritos en la sección anterior, la implementación del DWA de ROS, el planeador global por defecto de ROS que denominaremos NavFn, un planeador global basado en el algoritmo RRT\*, y los demás componentes de ROS necesarios en el proceso de navegación. Por otro lado, para los PICS en el DWA se incorpora un parámetro que se denomina  $P_{sc}$ , el cual es un valor de umbral que le permite al algoritmo determinar si un control es seguro o no, dependiendo de la calificación que obtuvo en la etapa de evaluación. Así todos los controles cuya calificación se encuentran arriba de  $P_{sc}$ , son considerados inseguros y por lo tanto son excluidos para la etapa de selección; así, al variar el valor de este parámetro cambia el nivel de seguridad en los controles seleccionados por el enfoque. Durante las pruebas se mantuvieron a los humanos fijos, pero se forzó a que los modelos de ocupación futura asumieran que el humano seleccionado se encontraba moviéndose en línea recta con una velocidad constante de 1 m/s. Esto se realizó con la finalidad de reproducir el efecto del PICS-DW en la navegación del robot ya que al tener el humano fijo en esa posición se crea siempre una zona de riesgo justo enfrente de él. Con los humanos en movimiento es difícil saber si el éxito al evitar una colisión es debido a la inclusión de los PICS o al efecto del modelo global de navegación.

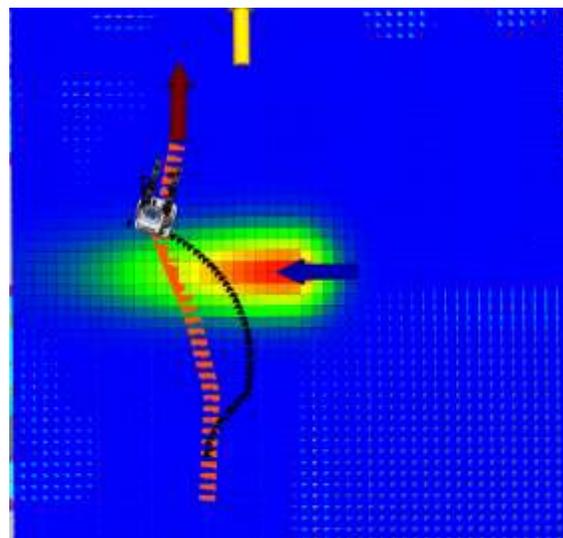
En la Figura 5 se muestra el comportamiento obtenido al usar los componentes propios de ROS, es decir a su planeador global RRT\* y al planeador local basado en el DWA.



**Figura 5** Algoritmo DWA en conjunto con RRT\*.  
Fuente: Elaboración Propia

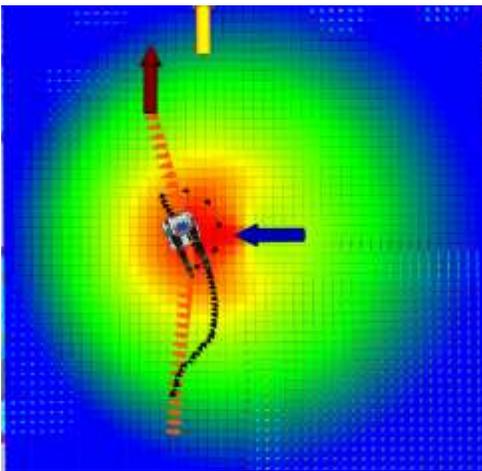
En esa imagen (y en todas las imágenes siguientes) las flechas rojas indican la posición y la orientación que se desea que alcance el robot (representa el estado objetivo), la flecha azul representa la posición y orientación del humano sobre el cual se centran las simulaciones, la línea naranja representa la trayectoria o plan proporcionado por el planeador global, la secuencia de flechas negras representa las distintas posiciones que ha realizado el robot debido a los controles seleccionados por el planeador local. En algunas imágenes aparece una tercera flecha de color amarillo que representa la posición y orientación del segundo humano presente en la simulación. La ejecución en la figura 5 muestra como en ausencia de algún conocimiento de la trayectoria futura del humano la trayectoria elegida se acerca a la posición del humano marcada en azul.

Las figuras siguientes fueron preparadas para resaltar el espacio ocupado por los modelos PS y el IPS; en tiempo de ejecución sólo se calculan los valores en los puntos de las trayectorias para evitar procesamiento innecesario. En la Figura 6 se puede observar que el robot pasa muy cerca de la posición real del humano, situación que lo pone en riesgo, incluso si se mantiene fijo en su posición. En tal configuración se utiliza el PICS-DW, con el modelo de ocupación PS y con un valor  $P_{sc}$  de 0.8. Se puede observar que con el planeador RRT\* el robot se tuvo que desviar bastante del plan global (en naranja) y eso causó que navegara cerca de un área marcada con probabilidades altas de colisión lo que claramente representa una situación de riesgo.



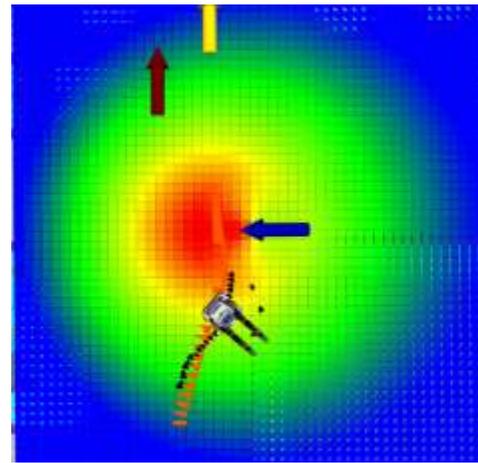
**Figura 6** Simulación con el algoritmo PICS-DW, el modelo PS y el planeador RRT\* con un valor de  $P_{sc}$  de 0.8  
Fuente: Elaboración Propia

Se analizará ahora lo que ocurre cuando se utiliza el PICS-DW con el modelo IPS y el planeador RRT\*. En la Figura 7 se muestran los resultados que se obtienen en simulación cuando se utiliza esta combinación con  $P_{sc} = 0.8$ . Se puede apreciar que el robot se queda rotando sobre su eje cerca del centro del área de ocupación mostrada donde las probabilidades son más altas. El motivo por el cual el robot termina entrando en la zona roja es porque el último control seleccionado para su ejecución fue evaluado como seguro para ese ciclo de ejecución y no es hasta el momento de la evaluación para la selección del control del siguiente ciclo de ejecución cuando se refleja el riesgo que impone el humano. Al utilizar el modelo IPS con  $P_{sc}=0.4$  se obtiene el comportamiento que se muestra en la Figura 8; en este caso se observa un cambio en el comportamiento ya que el robot se detiene antes de entrar en la zona roja mostrada. Esto se debe a que el control que causó que el robot se adentrara en esta zona ahora es descartado porque con el nuevo valor de  $P_{sc}$  el control es considerado inseguro. En este punto, los únicos controles seguros que el algoritmo puede encontrar son aquellos que implican rotar sobre su eje.



**Figura 7** Simulación con el algoritmo PICS-DW, el modelo IPS y el planeador RRT\* con un valor de  $P_{sc}$  de 0.8

*Fuente: Elaboración Propia*



**Figura 8** Simulación con el algoritmo PICS-DW, el modelo IPS y el planeador RRT\* con un valor de  $P_{sc}$  de 0.4

*Fuente: Elaboración Propia*

En la Tabla 1 se muestra un resumen de las pruebas realizadas y de los resultados obtenidos. La columna etiquetada como “Ruta Seguida” es una medida cualitativa de qué tan buena es la ruta real seguida por el robot (la que se representan por medio de una secuencia de flechas negras en las imágenes) donde la calificación de “Buena” indica que por lo general los movimientos realizados no se alejan mucho del plan global y tampoco cruzan por zonas de alto riesgo; la calificación de “Regular” indica que ocasionalmente pueden existir movimientos que se desvían un poco del plan global o que ocasionalmente atraviesan o invaden ligeramente zonas de alto riesgo; por último, la calificación de “Mala” indica que por lo general los movimientos se desvían demasiado del plan global o atraviesan zonas de alto riesgo. De manera similar, la columna etiquetada como “Seguridad” tiene una calificación de “Alta” cuando por lo general los movimientos realizados no atraviesan por zonas de alto riesgo; obtiene calificación “Media” cuando la mayoría de las veces los movimientos realizados logran evadir zonas de alto riesgo y obtiene calificación “Baja” cuando la mayoría de los movimientos no logra evitar las zonas de alto riesgo.

Planeador Global	Planeador Local	Modelo de Ocupación	Valor $P_{sc}$	Ruta Seguida	Seguridad
NavFn	DWA	N.A.	N.A.	Buena	Baja
RRT*	PICS-DWA	N.A.	N.A.	Buena	Baja
NavFn/RRT*	PICS-DWA	PS	0.8	Regular	Media
NavFn/RRT*	PICS-DWA	PS	0.4	Regular	Media
NavFn/RRT*	PICS-DWA	IPS	0.8	Regular	Media
NavFn/RRT*	PICS-DWA	IPS	0.4	Regular	Alta

**Tabla 1** Resumen de las distintas pruebas realizadas para el módulo de acciones seguras

Fuente: *Elaboración Propia*

Al analizar y comparar las simulaciones realizadas se observa que al utilizar el enfoque PICS-DW propuesto en este trabajo, existe un cambio en la manera en que se selecciona el control a ejecutar por el robot, de tal manera que el algoritmo rechaza aquellos controles que son considerados peligrosos y entre los controles seguros le da preferencia a aquellos que tienen un valor de probabilidad PICS bajo.

Las simulaciones demuestran también cómo al cambiar el modelo de ocupación futuro de los obstáculos se da un cambio en el comportamiento del robot al navegar, lo que confirma que el nivel de seguridad proporcionado por este enfoque depende del modelo de ocupación futuro. En el caso de los modelos presentados en este trabajo, se obtuvieron mejores resultados con el modelo basado en el IPS porque el robot evita cruzar delante de los humanos. Si durante la navegación el robot se acerca a un humano, lo suficiente para que sus sensores lo detecten, el enfoque rechazará los controles que intenten cruzar delante del humano (debido al modelo) y a los controles que intenten cruzar por encima del humano (debido a que los sensores detectan su presencia) por lo que sus principales opciones serán detenerse, rotar sobre su eje o cruzar por detrás del humano.

Por otro lado, a pesar de que el modelo basado en el PS es una mejor representación de la ocupación de los humanos, este modelo únicamente rechazará un control cuando, en un mismo instante de tiempo, tanto el humano como el robot se encuentren en posiciones muy cercanas, esto le permite al robot más opciones para navegar.

También se observó que durante las simulaciones que el enfoque propuesto funciona para cualquier planeador global, ya que, al final, la naturaleza de las rutas proporcionadas por el planeador global no influye en el desempeño del PICS-DWA. Se puede afirmar que el enfoque propuesto mejora el proceso de selección de los controles del DWA original al considerar los posibles movimientos de los humanos.

## Conclusiones y Trabajo Futuro

En este trabajo se realizó un estudio de enfoques de seguridad que pueden aplicarse a sistemas de navegación de robots, siendo el concepto de los Estados de Colisión Inevitable Probabilísticos el que se seleccionó para trabajar. Se propuso el enfoque de navegación reactiva PICS-DW el cual, incorpora el concepto de los Estados de Colisión Inevitable Probabilísticos (PICS) como una medida del riesgo de colisión al enfoque de las ventanas dinámicas (DWA). El enfoque demostró durante las pruebas, que es capaz de identificar, el nivel de riesgo que representa ejecutar cada uno de los controles considerados en la ventana dinámica y utilizar esta información para descartar aquellos que son peligrosos y priorizar aquellos que son seguros.

Bajo este enfoque, el cálculo de las probabilidades PICS se realiza con base en las posiciones actuales de los humanos y de los controles incorporados en la ventana dinámica; gracias a esto es posible realizar todo el proceso de forma on-line. A diferencia de los trabajos relacionados, no fue necesario calcular previamente las probabilidades PICS para todos los posibles estados que pudiera tomar el robot. El nivel de seguridad que puede ser alcanzado con este enfoque, depende en gran medida del modelo de ocupación empleado para los obstáculos; si el modelo es lo suficientemente robusto y preciso, el enfoque garantiza que no existirá una colisión al ejecutar el control elegido en la siguiente iteración. El sistema final queda disponible para que la comunidad pueda probar nuevas estrategias de navegación segura incorporando planeadores y modelos de robot diferentes. El enfoque aún requiere ser probado en escenarios más realistas, o en aplicaciones de la vida real, para confirmar su aplicabilidad ya que, a pesar de que se toma en cuenta un modelo matemático de movimiento de los humanos, tal modelo puede resultar muy restringido.

También resulta necesario el diseño e implementación de pruebas cuantitativas que permitan evaluar de mejor manera el rendimiento del enfoque. Es posible identificar algunas áreas de oportunidad específicas para continuar fortalecer esta propuesta, las cuales se listan a continuación:

- Extender la manera en que se representa la ventana dinámica ya que la actual no es válida para humanoides o hexápodos.
- Incorporar la representación del ambiente de 2D a 3D.
- Búsqueda y desarrollo de un mejor modelo de ocupación para los obstáculos que considere cambios de trayectoria e interacciones con otros humanos, objetos en el ambiente o el propio robot.
- Incorporar la capacidad de validar rutas a largo plazo de manera similar al presentado en (Axelrod, Kaelbling y Lozano-Pérez, 2018).
- Implementar un planeador que sea capaz de modificar una secuencia de controles, si es necesario, dependiendo de los cambios presentados en el ambiente.
- Comparar el presente enfoque con el presentado por (Lorente, Owen, y Montano, L., 2018).

Actualmente, estudiantes del Instituto Tecnológico de Mérida están realizando una extensión a esta propuesta para que pueda ejecutarse en un simulador de robots más completo y con soporte a más lenguajes de programación llamado VREP (por sus siglas en inglés, Virtual Experimentation Platform), propiedad de la empresa Coppelia Robotics.

### Agradecimientos

Se agradece al Consejo Nacional de Ciencia y Tecnología la contribución para el desarrollo de esta investigación mediante la beca 337663/273912 otorgada al primer autor.

### Referencias

- Alami, R., Krishna, K.M., y Simeon, T. (2007) *Provably Safe Motions Strategies for Mobile Robots in Dynamic Domains*, págs. 85-106. Springer Berlin Heidelberg
- Aoude, G.S., Luders, B.D., Joseph, J.M., Roy, N., y How, J.P. (2013) *Probabilistically Safe Motion Planning to Avoid Dynamic Obstacles with Uncertain Motion Patterns*. *Auton. Robots* 35(1): 51-76
- Axelrod, B., Kaelbling, L.P., Lozano-Pérez, T (2018). *Provably safe robot navigation with obstacle uncertainty*. *The International Journal of Robotics Research*, DOI: 10.1177/0278364918778338
- Bautin, A., Martinez-Gomez, L., y Fraichard, T. (2010) *Inevitable Collision States: A probabilistic perspective*. En *International Conference on Robotics and Automation*, págs. 4022-4027
- Bouraine, S., Fraichard, T., y Salhi, H. (2012) *Provably safe navigation for mobile robots with limited field-of-views in unknown dynamic environments*. En *International Conference on Robotics and Automation*, págs. 174-179
- Chung, W., Kim, S., Choi, M., Choi, J., Kim, H., b. Moon, C., y Song, J.B. (2009) *Safe Navigation of a Mobile Robot Considering Visibility of Environment*. En *IEEE Transactions on Industrial Electronics* 56(10):3941-3950
- Fiorini, P. y Shiller, Z. (1998) *Motion Planning in Dynamic Environments Using Velocity Obstacles*. *The International Journal of Robotics Research* 17(7):760-772. DOI:10.1177/027836499801700706
- Fox, D., Burgard, W., y Thrun, S. (1995) *The Dynamic Window Approach to Collision Avoidance*. Informe técnico
- ISO13482. (2014) *Robots and Robotic devices- Safety requirements for personal care robots*, International Organization for Standardization
- Karaman, S. y Frazzoli, (2011) *E. Sampling-based algorithms for optimal motion planning*. *The international journal of robotics research* 30(7):846-894
- PACHECO-FARFAN, L., RIOS-MARTINEZ, J, MORENO-SABIDO, M y SANDOVAL-CURMINA, V.. Implementación de un módulo de validación de acciones seguras para navegación de robots de servicio. *Revista de Ingeniería Innovativa* 2018

LaValle, S.M. (1998) Rapidly-exploring random trees: A new tool for path planning. Recuperado de:  
<http://msl.cs.illinois.edu/~lavelle/papers/Lav98c.pdf>

Lorente, M., Owen, E. y Montano, L. (2018) Model-based robocentric planning and navigation for dynamic environments. *The International Journal of Robotics Research*, Vol. 37(8) 867–889

Minguez, J. y Montano, L. (2004) Nearness diagram navigation: collision avoidance in troublesome scenarios. *IEEE Transactions on Robotics and Automation*, 20(1):45-59

Petti, S. y Fraichard, T. (2005) Safe motion planning in dynamic environments. *International Conference on Intelligent Robots and Systems*, 2210-2215

Rios-Martinez, J. (2013) *Socially-Aware Robot Navigation: combining Risk Assessment and Social Conventions*. Tesis, Université de Grenoble