

HuNavSim: simulador de navegación humana para evaluación de navegación social

Otero, R.^{a,*}, Pérez, N.^a, Caballero, F.^a, Merino, L.^a

^aEscuela Politécnica Superior, Universidad Pablo de Olavide, Crta. Utrera km 1, 41013, Sevilla, España.

To cite this article: Otero, R., Pérez, N., Caballero, F., Merino, L. 2023. HuNavSim: Human Navigation Simulator for Social Navigation Evaluation. XLIV Jornadas de Automática, 702-707. <https://doi.org/10.17979/spudc.9788497498609.702>

Resumen

En este trabajo se presenta el simulador *HuNavSim* ("Human Navigation Simulator"), una herramienta de código libre para la simulación de agentes humanos con comportamientos individualizados. La herramienta, la cual ha sido programada bajo el framework ROS2, puede ser usada junto a simuladores empleados en robótica como Gazebo. El objetivo principal de esta herramienta es facilitar el desarrollo y evaluación en simulación de sistemas de navegación de robots en espacios compartidos con humanos. Para ello, *HuNavSim* incluye un conjunto realista de comportamientos humanos y un completo conjunto de métricas para la evaluación de la navegación.

Palabras clave: Robótica, Sistemas Hombre-Máquina, Vehículos Autónomos Inteligentes.

HuNavSim: Human Navigation Simulator for Social Navigation Evaluation

Abstract

This work presents the Human Navigation Simulator (HuNavSim), an open-source tool for the simulation of different human-agent navigation behaviors in scenarios with mobile robots. The tool, programmed under the ROS 2 framework, can be employed along with different well-known robotics simulators like Gazebo. The main goal is to ease the development and evaluation of human-aware robot navigation systems in simulation. To do so, besides a general human- navigation model, HuNavSim includes a rich set of individual human navigation behaviors and a complete set of metrics for social navigation benchmarking.

Keywords: Robotics, Human Machine Systems, Intelligent Autonomous Vehicles

1. Introducción

La evaluación de las habilidades del robot para navegar en escenarios compartidos con humanos se está convirtiendo en algo esencial para la creación de robots de servicio. El desarrollo de estos robots sociales presenta principalmente, dos problemas: la experimentación real es muy costosa y difícil de llevar a cabo, a no ser que sean escenarios muy simples. Además, para realizar una prueba de un robot social, se requiere que haya personas en el escenario, lo que podría provocar situaciones de peligro para las mismas, sobre todo en las fases de prueba iniciales. Por tanto, llevar a cabo una simulación para la navegación que cuente con comportamientos humanos realistas es ne-

cesario. En segundo lugar, cuando se evalúa la navegación de un robot social no solo es importante considerar su eficiencia, sino también la seguridad y comodidad de las personas. Este último requisito, la comodidad, se refiere a sensaciones subjetivas, lo que dificulta modelarlo mediante ecuaciones matemáticas. Por este motivo, no existe un acuerdo sólido entre la comunidad investigadora sobre un conjunto de métricas válido para dicho propósito.

La mayoría de soluciones actuales para la simulación de personas están basadas en modelos de movimiento de multitudes para controlar el comportamiento de los agentes humanos. Esto es válido para obtener un comportamiento colectivo de los

*Autor para correspondencia: rotegal@alu.upo.es
Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0)

agentes, pero pierde realismo a un nivel más individualizado, ya que el comportamiento para todos los agentes es exactamente el mismo en todos los escenarios. Por tanto, proponemos un conjunto de comportamientos humanos individualizados asociados a la presencia del robot.

Otro problema es la evaluación de la navegación social a través de métricas. Cada herramienta de evaluación presenta su propio conjunto de métricas. La investigación de nuevas métricas de navegación social realistas es deseable, ya que la ausencia de un conjunto de métricas consensuado dificulta la comparación de las técnicas de navegación social.

Con HuNavSim tenemos el objetivo de contribuir con soluciones para los dos problemas mencionados anteriormente: proporcionando un conjunto de diferentes comportamientos realistas individualizados para los agentes, y presentando una recopilación completa de métricas usadas en la literatura. En resumen, se presentan las siguientes contribuciones:

- i) Una herramienta flexible y de código abierto para la navegación de humanos bajo el framework ROS2 (Macenski et al., 2022), que puede ser usado con diferentes simuladores de robótica.
- ii) Un conjunto de comportamientos para la navegación de los humanos, la cual incluye un conjunto realista de reacciones individuales ante la presencia de un robot.
- iii) Una completa recopilación de métricas recogidas de la literatura para la evaluación de robots sociales, la cual es configurable y se puede extender.
- iv) Una interfaz para usar la herramienta con el simulador Gazebo, que es usado en el ámbito de la robótica. (véase la Figura. 1).

2. Estado del arte

Actualmente, existen varios simuladores y herramientas de evaluación para la navegación social en entornos compartidos con personas. A continuación, presentamos una revisión exhaustiva de todo el software que está relacionado con nuestro enfoque, destacando las diferencias y similitudes entre ellos.

*PedSimROS*¹ y *MengeROS*² (Aroor et al., 2017) son simuladores de multitudes desarrollados bajo versiones obsoletas de ROS1. Además, no incorporan ninguna opción para la evaluación de la navegación.

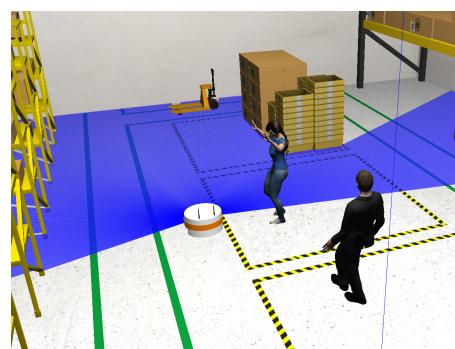


Figura 1: Captura de los agentes del simulador en Gazebo

*CrowdBot*³ (Grzeskowiak et al., 2021) y *SEAN (Social Environment for Autonomous Navigation)*⁴ (Tsoi et al., 2022, 2020) son ejemplos de herramientas más recientes, avanzadas y ambiciosas. Estas herramientas comparten características similares ya que ambas están basadas en el motor de juegos Unity y ROS1; y su objetivo principal es convertirse en la herramienta estándar para la evaluación de la navegación de robots en entornos con mucha población. En contraste con estas herramientas, *HuNavSim* ofrece una solución más flexible al permitir su uso con diversos simuladores y proporcionar un conjunto de comportamientos humanos individuales y realistas en presencia de un robot. Mientras que estas herramientas presentan un conjunto fijo de métricas, *HuNavSim* incluye una recopilación de métricas fácilmente configurable y ampliable.

Otro simulador interesante es *Intelligent Human Simulator (InHuS)* (Favier et al., 2021b,a). Este simulador tiene como objetivo controlar el movimiento de los agentes humanos simulados en otro simulador. Al igual que *HuNavSim*, también incorpora un conjunto reducido de comportamientos humanos individuales. Sin embargo, está basado en ROS1 y utiliza principalmente un sistema de navegación diseñado para robots llamado *HATEB2* (Teja S. and Alami, 2020) para gestionar el movimiento de los agentes humanos. Esto podría resultar en un movimiento de multitudes menos realista en comparación con simuladores que cuentan con modelos de movimiento específicos para tal propósito.

*SocNavBench*⁵ (Biswas et al., 2022) es una herramienta de evaluación para simuladores que utiliza información previamente grabada de humanos reales, que se reproduce durante la ejecución de una simulación. Sin embargo, su principal inconveniente radica en que las trayectorias de los humanos se reproducen a partir de conjuntos de datos, lo que implica que los efectos causados por el robot en el camino de los humanos no son considerados. Esto dificulta la obtención de una evaluación realista de navegación social.

Por último, cabe mencionar herramientas de evaluación muy interesantes como *BARN (Benchmark for Autonomous Robot Navigation)*⁶ (Perille et al., 2020) y *Bench-MR*⁷ (Heiden

¹https://github.com/srl-freiburg/pedsim_ros

²https://github.com/ml-lab-cuny/menge_ros

³<http://crowdbot.eu/CrowdBot-challenge/>

⁴<https://sean.interactive-machines.com/>

⁵<https://github.com/CMU-TBD/SocNavBench>

⁶https://www.cs.utexas.edu/~atruong/metrics_dataset.html

⁷<https://github.com/robot-motion/bench-mr>

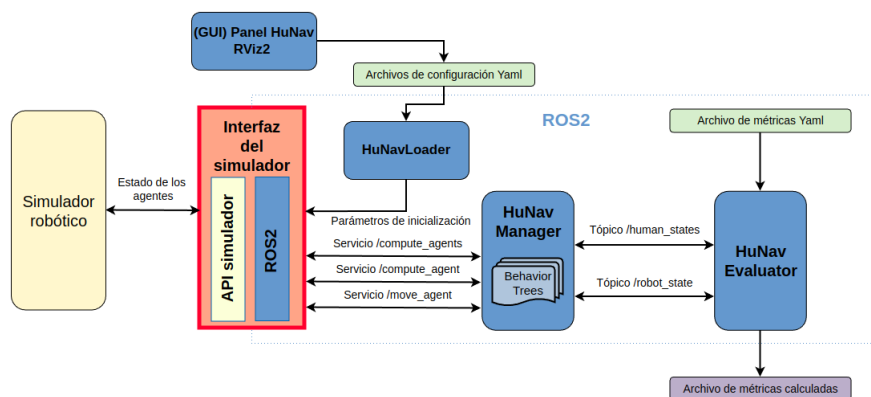


Figura 2: Diagrama del simulador. En azul, se encuentran los módulos de *HuNavSim*. En amarillo, los módulos del simulador robótico base. En verde, se indican los archivos de configuración que contienen los datos de entrada. El resultado de la evaluación está indicado en violeta.

et al., 2021). Sin embargo, estas herramientas están principalmente enfocadas en el problema general de la navegación en entornos desordenados sin considerar el componente social presente en entornos compartidos con humanos.

3. Simulador de Navegación Humana (HuNavSim)

3.1. Arquitectura del simulador

La Figura 2 muestra la arquitectura general del simulador. *HuNavSim* desempeña el papel de controlar los agentes humanos generados en otro simulador base, como Gazebo, Morse, o Webots, que también deben simular el escenario y el robot. Para lograr esto, es necesario contar con una interfaz que facilite la comunicación entre el simulador base y *HuNavSim*.

En primer lugar, el usuario debe proporcionar la cantidad de agentes y los parámetros de configuración necesarios. Estos pueden especificarse a través de un archivo de configuración yaml o mediante una interfaz gráfica de usuario basada en un panel de RViz de ROS2.

Una vez que se ha proporcionado la información de los agentes, la interfaz del simulador envía el estado de cada agente en cada paso de tiempo al módulo *hunav_manager* a través de los servicios de ROS2. El sistema determina el siguiente estado de los agentes en función de su estado actual, devolviendo posteriormente los nuevos estados a la interfaz, que los actualiza en el simulador base.

Finalmente, el módulo *hunav_evaluator* registra los datos que se van generando durante el experimento y realiza el cómputo de las métricas al final de cada simulación. La información sobre cada simulación y el conjunto de métricas que se desea calcular puede especificarse mediante un archivo de configuración yaml o a través de otra interfaz gráfica basada también en un panel de RViz. Una vez que se completa la simulación y se han calculado las métricas, este módulo genera un archivo que contiene la información de la simulación y los resultados obtenidos para cada métrica.

La documentación completa y el código de *HuNavSim* están disponibles en: https://github.com/robotics-upo/hunav_sim

3.2. Modelo de navegación humana

HuNavSim está basado principalmente en el uso del Modelo de Fuerzas Sociales (*SFM*) (Helbing and Molnár, 1995) y su extensión para grupos de personas (Moussaïd et al., 2009; Moussaïd et al., 2010), que utiliza un conjunto de fuerzas de atracción y repulsión para guiar a los agentes durante la navegación, ayudándoles a evitar obstáculos y alcanzar sus objetivos de manera eficiente. Sin embargo, hemos ampliado este modelo para ofrecer un conjunto de reacciones individualizadas y realistas para la navegación de los agentes en presencia del robot. Esto enriquece los escenarios de navegación y desafía a los algoritmos de navegación con comportamientos humanos más diversos y realistas. El conjunto de comportamientos contiene las siguientes reacciones:

- *Normal*: el agente humano trata al robot como otro humano.
- *Impasible*: el agente humano trata al robot como un obstáculo estático.
- *Sorprendido*: cuando el agente humano ve al robot, detiene su movimiento y comienza a observar al robot.
- *Curioso*: el agente humano abandona el objetivo de navegación que tiene asignado y comienza a aproximarse al robot lentamente.
- *Miedoso*: el agente humano intenta mantenerse alejado del robot.
- *Amenazante*: el agente humano intenta bloquear el movimiento del robot para evitar que pueda continuar su camino.

Una característica interesante es que todos los comportamientos son controlados por árboles de comportamiento (Colledanchise and Ögren, 2018). Estos árboles permiten una estructura eficiente para efectuar el cambio entre diferentes tareas o acciones de los agentes autónomos. Además, son fáciles de programar y se pueden agregar o modificar árboles de comportamiento sin dificultad. Específicamente, utilizamos el motor *BehaviorTree.CPP*⁸ para este propósito.

⁸<https://github.com/BehaviorTree/BehaviorTree.CPP>

3.3. Interfaz para el simulador Gazebo

Se ha desarrollado una interfaz para utilizar *HuNavSim* junto al simulador Gazebo. En la Figura 3 se presentan los módulos que componen esta interfaz, así como los módulos de comunicación con Gazebo y *HuNavSim*.

El controlador de movimiento de los agentes ha sido implementado como un plugin para Gazebo, el cual debe ser incluido en el archivo de configuración del mundo de Gazebo junto con los agentes, que son incluidos como actores de Gazebo. Para facilitar esta tarea, se ha desarrollado un módulo llamado *world_generator*, el cual se encarga de leer los parámetros de los agentes y escribir el plugin y los agentes en el archivo de configuración del mundo de Gazebo. Este plugin establece la comunicación con *HuNavSim* para mantener actualizado el estado de los agentes.

Además, se ha incluido un conjunto de escenarios típicos que suelen involucrar interacciones con humanos, como una cafetería, un almacén y una casa.

Toda la documentación y el código relacionados con esta interfaz está disponible en: https://github.com/robotics-upo/hunav_gazebo_wrapper

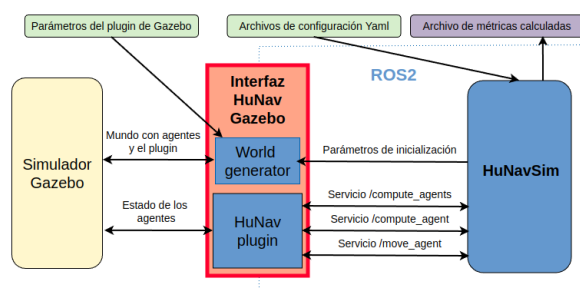


Figura 3: Diagrama del interfaz HuNavSim-Gazebo

4. Métricas

Con el objetivo de abordar el problema de la selección de las mejores métricas más adecuadas para la evaluación de la navegación social, nos hemos esforzado por mantener el sistema de evaluación lo más flexible posible.

En primer lugar, realizamos una revisión exhaustiva de la literatura para recopilar la mayoría de métricas aplicadas a este problema. Además, permitimos que los usuarios sean quienes elijan qué métricas desean calcular en cada simulación. Hemos diseñado el sistema de manera que sea sencillo añadir nuevas métricas al conjunto predefinido. Estas métricas reciben dos vectores que contienen las posiciones, velocidades, y otra información relevante de los agentes y el robot para cada paso de tiempo.

De esta manera, *HuNavSim* ofrece una forma flexible y de calidad para el sistema de evaluación en comparación con los sistemas encontrados en la literatura.

Las métricas que están implementadas actualmente son aquellas usadas en nuestro trabajo anterior (Perez-Higueras et al., 2018), del simulador SEAN (Tsoi et al., 2022), algunas de SocNavBench (Biswas et al., 2022), (Katyál et al., 2022) y del SFM (Helbing and Molnár, 1995). La Tabla 1 muestra la lista de métricas que conforman el conjunto de evaluación.

5. Experimentos

Según Gao *et al.* (Gao and Huang, 2022), los escenarios comúnmente utilizados en la evaluación de la navegación social son: Cruce en direcciones opuestas, Cruce en direcciones perpendiculares, Adelantamiento, Aproximación, Seguir/Acompañar y Combinaciones entre las anteriores. En esta evaluación, hemos realizado una evaluación en un escenario de Cruce en direcciones opuestas, en el cual el objetivo del robot es llegar al extremo opuesto de la habitación mientras se encuentra con dos personas caminando en dirección contraria.

En el escenario anterior, se realizará una comparación de las métricas calculadas obtenidas a partir de tres planificadores que poseen con diferentes características:

- *DWB*. Se trata de la pila de navegación fundamental de ROS2 que emplea el planificador DWB, el cual se basa principalmente en el enfoque de la Ventana Dinámica (Fox et al., 1997).
- *SCL*. Este enfoque implica la utilización del mismo algoritmo mencionado anteriormente, pero incorporando una capa social en el mapa de costes.
- *SFW*. un planificador social, desarrollado por nosotros, que se basa en el enfoque de la Ventana Dinámica. Este planificador utiliza el Modelo de Fuerza Social para predecir la posición de los agentes sin conocimiento previo del comportamiento de estos, además, emplea una función para puntuar la trayectoria que tiene consideraciones sociales.

Para llevar a cabo la evaluación, se repitió la trayectoria diez veces para cada planificador. Por tanto, los valores mostrados en la Tabla 2 corresponden al promedio de cada métrica.

A partir de estos resultados, podemos obtener una serie de conclusiones. Es evidente que el controlador DWB carece de la consideración de cualquier componente social, dado que su objetivo primordial es alcanzar el objetivo final de manera rápida y evitar colisiones, tratando a todas las personas como simples obstáculos. En segundo lugar, el SCL añade un coste extra al mapa de navegación alrededor de las personas, resultando en que el robot mantenga una mayor distancia respecto a las personas. Por último, el planificador SFW incorpora restricciones sociales. Concretamente, emplea el trabajo social como un factor en su función de puntuación de trayectoria. Esto facilita al robot mantener de manera más efectiva la coherencia con las personas que lo rodean.

6. Conclusiones y trabajo futuro

Hemos presentado de manera concisa un nuevo software de código abierto para la simulación de navegación de humanos con comportamientos. Este sistema, desarrollado bajo el framework ROS2, permite controlar los agentes humanos en diferentes simuladores de uso general. Además, incluye características innovadoras como un conjunto realista de comportamientos individuales para humanos, controlados por árboles de comportamiento, y una completa y adaptable recopilación de métricas para la evaluación.

Tabla 1: Métricas disponibles en *HuNavSim*

Métricas	Referencias	Descripción	Unidades
Tiempo en llegar al objetivo final	(Perez-Higueras et al., 2018)	Tiempo en llegar al objetivo final	<i>s</i>
Longitud del camino	(Perez-Higueras et al., 2018)	Longitud del camino realizado por el robot	<i>m</i>
Cambios de ángulo de dirección acumulado	(Perez-Higueras et al., 2018)	Sumatorio en valor absoluto de la variación del ángulo de la dirección en la trayectoria	<i>rad</i>
Distancia media a la persona más cercana	(Perez-Higueras et al., 2018)	Media de la distancia a la persona más cercana en cada momento	<i>m</i>
Distancia mínima a las personas	(Perez-Higueras et al., 2018)	Valor mínimo de las distancias entre el robot y las personas durante la trayectoria	<i>m</i>
Intrusiones en el espacio íntimo	(Perez-Higueras et al., 2018)	Tiempo que el robot ha estado en el espacio íntimo de una persona (Hall, 1990)	%
Intrusiones en el espacio personal	(Perez-Higueras et al., 2018)	Tiempo que el robot ha estado en el espacio personal de una persona (Hall, 1990)	%
Intrusiones en el espacio social	(Perez-Higueras et al., 2018)	Tiempo que el robot ha estado en el espacio social de una persona (Hall, 1990)	%
Intrusiones en el espacio íntimo de grupos	(Perez-Higueras et al., 2018)	Tiempo que el robot ha estado en el espacio íntimo de un grupo	%
Intrusiones en el espacio personal de grupos	(Perez-Higueras et al., 2018)	Tiempo que el robot ha estado en el espacio personal de un grupo	%
Intrusiones en el espacio social de grupos	(Perez-Higueras et al., 2018)	Tiempo que el robot ha estado en el espacio personal de un grupo	%
completado	(Tsoi et al., 2022)	Indica si ha llegado al objetivo final o no	-
Distancia mínima al objetivo	(Tsoi et al., 2022)	Distancia mínima a los objetivos que ha seguido el robot durante la trayectoria	<i>m</i>
Distancia final al objetivo	(Tsoi et al., 2022)	Distancia entre la última posición del robot y el objetivo final.	<i>m</i>
Colisiones del robot con personas	(Tsoi et al., 2022)	Número de veces que el robot colisionó contra una persona	-
Colisiones de personas con el robot	(Tsoi et al., 2022)	Número de veces que una persona colisionó contra el robot	-
Tiempo sin moverse	(Tsoi et al., 2022)	Computa los segundos que el robot ha estado sin movimiento	<i>s</i>
Velocidad lineal media del robot	(Biswas et al., 2022)	Computa la velocidad lineal media del robot durante la trayectoria	<i>m/s</i>
Velocidad angular media del robot	(Biswas et al., 2022)	Computa la velocidad angular media en valor absoluto del robot durante la trayectoria	<i>rad/s</i>
Aceleración media del robot	(Biswas et al., 2022)	Computa la media de la aceleración lineal en valor absoluto, teniendo en cuenta los valores de aceleración y deceleración	<i>m/s²</i>
Sobre aceleración media del robot	(Biswas et al., 2022)	Computa la media de sobre aceleración que ha experimentado el robot durante la trayectoria	<i>m/s³</i>
Velocidad lineal media de las personas	(Katyal et al., 2022)	Computa la velocidad lineal media de todos los agentes	<i>m/s</i>
Velocidad lineal media del agente más cercano	(Katyal et al., 2022)	Computa la velocidad lineal media del agente más cercano al robot	<i>m/s</i>
Fuerza social en agentes	(Helbing and Molnár, 1995)	Sumatorio del módulo de la fuerza social provocada por el robot en los agentes	<i>m/s²</i>
Fuerza social en el robot	(Helbing and Molnár, 1995)	Sumatorio del módulo de la fuerza social provocada por los agentes en el robot	<i>m/s²</i>
Fuerza de obstáculos en agentes	(Helbing and Molnár, 1995)	Sumatorio del módulo de la fuerza provocada por los obstáculos en los agentes	<i>m/s²</i>
Fuerza de obstáculos en el robot	(Helbing and Molnár, 1995)	Sumatorio del módulo de la fuerza provocada por los obstáculos en el robot	<i>m/s²</i>
Trabajo social	(Helbing and Molnár, 1995)	Es la suma de la fuerza social en el robot, la fuerza ejercida por los obstáculos en el robot y la fuerza social ejercida en los agentes	<i>m/s²</i>

En cuanto al trabajo futuro, está previsto completar todo el conjunto de métricas de evaluación. Actualmente, estamos estudiando y planeamos incluir las métricas de *Crowdbot* (Grzeskowiak et al., 2021) y las mencionadas en el trabajo de Gao et al. (Gao and Huang, 2022). El desarrollo de interfaces para otros simuladores como Webots o Isaac Sim también está siendo considerado actualmente. Además, se añadirán nuevos comportamientos de navegación para los humanos con el fin de ampliar las reacciones disponibles. Asimismo, se añadirán otros modelos de navegación humana aparte del Modelo de Fuerzas Sociales.

Tabla 2: Métricas para el escenario Cruce en direcciones opuestas

Cruce en direcciones opuestas	DWB	SCL	SFW	Unidades
Tiempo en llegar al objetivo final	25,5	26,6	35,7	<i>s</i>
Longitud del camino	15,2	15,7	15,2	<i>m</i>
Cambios de ángulo de dirección acumulado	0,47	1,89	1,89	<i>rad</i>
Distancia media a la persona más cercana	6,36	7,95	7,49	<i>m</i>
Distancia mínima a las personas	0,40	0,90	0,55	<i>m</i>
Intrusiones en el espacio íntimo	3,66	0	0	<i>%</i>
Intrusiones en el espacio personal	19,7	5,8	14,7	<i>%</i>
Intrusiones en el espacio social	17,0	26,7	21,4	<i>%</i>
completado	100	100	100	<i>%</i>
Distancia mínima al objetivo	0,18	0,19	0,11	<i>m</i>
Distancia final al objetivo	0,18	0,19	0,11	<i>m</i>
Colisiones del robot con personas	0	0	0	-
Colisiones de personas con el robot	0	0	0	-
Tiempo sin moverse	0,00	1,04	1,78	<i>s</i>
Velocidad lineal media del robot	0,51	0,51	0,38	<i>m/s</i>
Velocidad angular media del robot	0,02	0,09	0,09	<i>rad/s</i>
Aceleración media del robot	0,05	0,05	0,08	<i>m/s²</i>
Sobre aceleración media del robot	0,08	0,06	0,08	<i>m/s³</i>
Velocidad lineal media de las personas	0,47	0,44	0,34	<i>m/s</i>
Velocidad lineal media del agente cercano	0,57	0,51	0,39	<i>m/s</i>
Fuerza social en agentes	0,52	0,97	0,49	<i>m/s²</i>
Fuerza social en el robot	0,52	0,92	0,49	<i>m/s²</i>
Fuerza de obstáculos en agentes	8,14	0,13	0,81	<i>m/s²</i>
Fuerza de obstáculos en el robot	0,00	0,00	0,00	<i>m/s²</i>
Trabajo social	1,04	1,89	0,98	<i>m/s²</i>

Agradecimientos

Este trabajo ha sido parcialmente financiado por el proyecto NHOA PLEC2021-007868 subvencionado por MCI-N/AEI/10.13039/501100011033 y la Unión Europea NextGenerationEU/PRTR, y por el proyecto NORDIC TED2021-132476B-I00, subvencionado por el Ministerio de Ciencia.

Referencias

Aroor, A., Epstein, S. L., Korpan, R., 2017. Mengers: a crowd simulation tool for autonomous robot navigation. In: AAAI Fall Symposia.
 Biswas, A., Wang, A., Silvera, G., Steinfeld, A., Admoni, H., jul 2022. Socnavbench: A grounded simulation testing framework for evaluating social navigation. ACM Transactions on Human-Robot Interaction.

URL: <https://doi.org/10.1145/3476413>
 DOI: 10.1145/3476413
 Colledanchise, M., Ögren, P., 2018. Behavior Trees in Robotics and AI: An Introduction. Chapman & Hall/CRC Artificial Intelligence and Robotics Series. CRC Press, Boca Raton, US.
 DOI: <https://doi.org/10.1201/9780429489105>
 Favier, A., Singamaneni, P.-T., Alami, R., Jun. 2021a. An Intelligent Human Simulation (InHuS) for developing and experimenting human-aware and interactive robot abilities, working paper or preprint.
 Favier, A., Singamaneni, P.-T., Alami, R., Jul. 2021b. Simulating Intelligent Human Agents for Intricate Social Robot Navigation. In: RSS Workshop on Social Robot Navigation 2021. Washington, United States.
 Fox, D., Burgard, W., Thrun, S., 1997. The dynamic window approach to collision avoidance. IEEE Robotics & Automation Magazine 4 (1), 23–33.
 DOI: 10.1109/100.580977
 Gao, Y., Huang, C.-M., 2022. Evaluation of socially-aware robot navigation. Frontiers in Robotics and AI 8.
 URL: <https://www.frontiersin.org/articles/10.3389/frobt.2021.721317>
 DOI: 10.3389/frobt.2021.721317
 Grzeskowiak, F., Gonon, D., Dugas, D., Paez-Granados, D., Chung, J. J., Nieto, J., Siegwart, R., Billard, A., Babel, M., Pettré, J., 2021. Crowd against the machine: A simulation-based benchmark tool to evaluate and compare robot capabilities to navigate a human crowd. In: 2021 IEEE International Conference on Robotics and Automation (ICRA). pp. 3879–3885.
 DOI: 10.1109/ICRA48506.2021.9561694
 Hall, E. T., Oct. 1990. The Hidden Dimension. Anchor.
 Heiden, E., Palmieri, L., Bruns, L., Arras, K. O., Sukhatme, G. S., Koenig, S., 2021. Bench-mr: A motion planning benchmark for wheeled mobile robots. IEEE Robotics and Automation Letters 6 (3), 4536–4543.
 DOI: 10.1109/LRA.2021.3068913
 Helbing, Molnár, 1995. Social force model for pedestrian dynamics. Physical review. E, Statistical physics, plasmas, fluids, and related interdisciplinary topics 51 5, 4282–4286.
 Katyal, K., Gao, Y., Markowitz, J., Pohland, S., Rivera, C., Wang, I.-J., Huang, C.-M., 2022. Learning a group-aware policy for robot navigation. In: 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 11328–11335.
 DOI: 10.1109/IROS47612.2022.9981183
 Macenski, S., Foote, T., Gerkey, B., Lalancette, C., Woodall, W., 2022. Robot operating system 2: Design, architecture, and uses in the wild. Science Robotics 7 (66), eabm6074.
 URL: <https://www.science.org/doi/abs/10.1126/scirobotics.abm6074>
 DOI: 10.1126/scirobotics.abm6074
 Moussaïd, M., Perozo, N., Garnier, S., Helbing, D., Theraulaz, G., 2010. The walking behaviour of pedestrian social groups and its impact on crowd dynamics. PLoS One 5 (4).
 DOI: 10.1371/journal.pone.0010047
 Moussaïd, M., Helbing, D., Garnier, S., Johansson, A., Combe, M., Theraulaz, G., August 2009. Experimental study of the behavioural mechanisms underlying self-organization in human crowds. Proceedings. Biological sciences 276 (1668), 2755–2762.
 DOI: 10.1098/rspb.2009.0405
 Perez-Higueras, N., Caballero, F., Merino, L., 2018. Teaching Robot Navigation Behaviors to Optimal RRT Planners. International Journal of Social Robotics 10 (2), 235–249.
 DOI: 10.1007/s12369-017-0448-1
 Perille, D., Truong, A., Xiao, X., Stone, P., 2020. Benchmarking metric ground navigation. In: 2020 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR). Abu Dhabi, UAE, pp. 116–121.
 Teja S., P., Alami, R., 2020. Hateb-2: Reactive planning and decision making in human-robot co-navigation. In: 2020 29th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN). pp. 179–186.
 DOI: 10.1109/RO-MAN47096.2020.9223463
 Tsoi, N., Hussein, M., Espinoza, J., Ruiz, X., Vázquez, M., 11 2020. Sean: Social environment for autonomous navigation. In: the 8th International Conference on Human-Agent Interaction (HAI '20). pp. 281–283.
 DOI: 10.1145/3406499.3418760
 Tsoi, N., Xiang, A., Yu, P., Sohn, S. S., Schwartz, G., Ramesh, S., Hussein, M., Gupta, A. W., Kapadia, M., Vázquez, M., 2022. Sean 2.0: Formalizing and generating social situations for robot navigation. IEEE Robotics and Automation Letters 7 (4), 11047–11054.
 DOI: 10.1109/LRA.2022.3196783