

Robótica asistencial y su interacción con entorno en oficinas

Mario Ricardo Arbulú*

Resumen

En este artículo se describen los fundamentos de los algoritmos de navegación autónoma y manipulación empleados para dotar de autonomía al Robot Asistencial de la CUN y que pueda ser empleado como asistente en oficinas. Los algoritmos de navegación se basan en técnicas del lugar de las raíces que, mediante la selección de polos y ceros en la zona de movimiento del robot, generan una trayectoria libre de obstáculos, lo que permite que el robot se desplace de un lugar a otro sin problemas. Para la manipulación de objetos, se plantean algoritmos de movimiento de brazos, cuando el robot está lo suficientemente cerca al objetivo con el que debe trabajar, basados en los parámetros de Denavit-Hartenberg; ahora bien, estos son modificados hacia la evaluación del espacio de trabajo aumentado de los brazos y el uso de trayectorias cartesianas suaves, estas últimas generadas a partir de la configuración del objeto que se desea alcanzar. Gracias a la presentación y discusión de los resultados, es posible concluir que la implementación del robot asistencial es viable y válida.

Palabras clave: cinemática, navegación autónoma, parámetros D-H, planificación de movimiento, robótica asistencial, robótica de servicios, visión artificial

Abstract

This paper describes the fundamentals of the autonomous navigation and manipulation algorithms used to give the CUN Assistance Robot autonomy and that it can be used as an office assistant. The navigation algorithms are based on root locus techniques that, by selecting poles and zeros in the robot's motion zone, generate an obstacle-free path, allowing the robot to move from one place to another. For object manipulation, arm movement algorithms are proposed, when the robot is close enough to the target with which it must work, based on the Denavit-Hartenberg parameters; however, these are modified towards the evaluation of the increased working space of the arms and the use of smooth Cartesian trajectories, the latter generated from the configuration of the object to be reached. Thanks to the presentation and discussion of the results, it is possible to conclude that the implementation of the assistance robot is viable and valid.

Keywords: artificial vision, assistive robotics, autonomous navigation, D-H parameters, kinematics, movement planning, service robotics

* Ph. D. en Ingeniería Eléctrica, Electrónica y Automática de la Universidad Carlos III de Madrid. Docente investigador del programa de Ingeniería Electrónica en la Corporación Unificada Nacional de Educación Superior (CUN). Contacto: mario_arbulu@cun.edu.co

Introducción

En la actualidad, los robots asistenciales se emplean como elementos de compañía a personas mayores, terapeutas de niños autistas, sirviente de herramientas o elementos de distinta índole y en diferentes contextos, como oficinas, hogares, salas de cirugía o eventos públicos, entre otros. Los robots referentes en esta línea son Peper, de la empresa japonesa Softbank, y el REEM de PAL Robotics: el primero, dotado con inteligencia artificial para reconocer gestos, hacer sugerencias, acompañar a personas, ser anfitrión en eventos

y trabajar en investigación y desarrollo; el segundo, un robot más robusto mecánicamente hablando y con un coste muy superior, aunque se emplea en tareas similares. En vista de que en las investigaciones del grupo IDECUN se pretende introducir estas temáticas de punta, se propone el presente proyecto, que consiste en integrar dos plataformas estándares que se complementan en cuanto a la mecánica, interacción con el usuario, entre otros (Kemp, Edsinger y Torres, 2007; Kruse, Radke y Wen, 2015).

Figura 1. Robot asistencial.



Fuente: elaboración propia.

El robot asistencial de la Corporación Unificada Nacional de Educación Superior (CUN) consta de dos partes, como se ve en la figura 1: de un lado, un manipulador redundante (humanoide) y, del otro, locomoción móvil omnidireccional (robot con ruedas). Su desarrollo se dio en el marco

de la convocatoria interna de proyectos de investigación de la CUN en año 2014, iniciando su ejecución a finales de 2015. El objetivo de este dispositivo es que sirva como asistente en hogares y oficinas.

La parte manipuladora, que está dotada de brazos, piernas, sensores ultrasónicos, cámaras y sensores táctiles, se encarga de coger objetos y dar las indicaciones necesarias, mientras que la tarea de la parte locomotora, que está dotada de

tres ruedas omnidireccionales, sensores ultrasónicos y sensores de contacto, es movilizar al robot por la zona del hogar u oficina en la que se requieran sus servicios de asistencia. Las especificaciones se describen en la tabla 1.

Tabla 1. Especificaciones del robot asistencial de la CUN.

Unit	Feature	Specifications
Redundant manipulator (NAO™)	Height	58 cm
	Weight	5.4 kg
	Autonomy	90 min
	DOF	25
	CPU	Intel Atom @ 1.6 GHz
	Built-in OS	NAOqi 2.0 (Linux-based)
	Compatible OS	Windows, MAC OS, Linux
	Programming languages	C++, Python, Java, MATLAB, Urbi, C, .Net
	Sensors	3 HD cameras, 4 microphones, sonar rangefinder, 2 infrared emitters and receivers, inertial unit and 9 tactile sensors.
	Connectivity	8 pressure sensors WiFi, Ethernet
Wheeled locomotion (Robotino™)	Height	29 cm
	Weight	5 kg
	Diameter	45 cm
	Used CPU	Intel Core i5 2.4 GHz, 8 GB RAM
	Compatible OS	Windows
	Programming languages	C/C++, Java, .Net, LabVIEW, MATLAB, Simulink, ROS, Microsoft Rob. Developer St.
	Sensors	9 distance infrared sensors, inductive sensor, 2 optical sensors and Full HD camera
	Connectivity	2 Ethernet, 6 USB, 1 VGA y 2 PCI

Fuente: elaboración propia.

Fundamentos teóricos

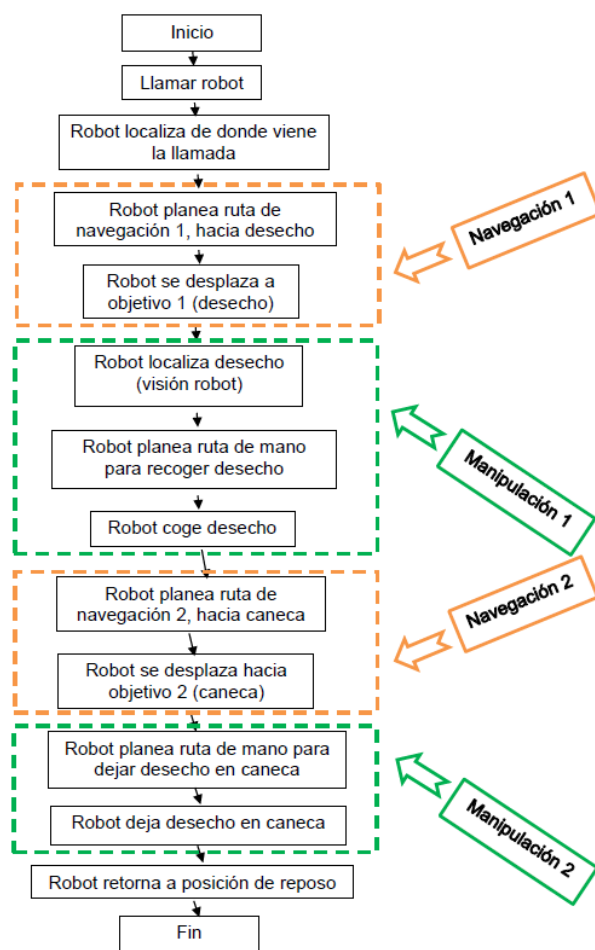
En esta sección se describe la fundamentación fisicomatemática de los algoritmos para el desarrollo de tareas autónomas, asistidas por visión artificial, que son aplicados en el robot asistencial de la CUN. En la figura 2 se detalla el caso de una tarea autónoma general; por ejemplo, para

que recoja un objeto, en cualquier parte de la oficina o zona de trabajo, y luego lo lleve hasta una caneca. Como se muestra, hay dos subtareas por desarrollar: 1) navegación autónoma y 2) manipulación autónoma. Las dos son asistidas por visión artificial: la primera, por una cámara

externa al robot, situada en la parte superior de su desplazamiento, y, la segunda, por las cámaras que el robot tiene en la cabeza.

En este documento explicamos cómo se calcula, primero, el movimiento del manipulador redundante (subtarea de manipulación autónoma) (Hayne, Luo y Berenson, 2016) y, luego, la ruta de navegación (subtarea de navegación autónoma).

Figura 2. Diagrama de flujo de las subtareas que debe desarrollar el robot asistencial para completar una tarea.



Fuente: elaboración propia.

Manipulación

La manipulación de objetos se realizará gracias a la generación de trayectorias de movimiento del humanoide, una vez esté lo suficientemente cerca del objeto en cuestión. Estas trayectorias se generarán a partir de la referencia de posición y orientación espacial del objeto por manipular.

Ahora bien, acá vale la pena aclarar que esta referencia se obtiene del algoritmo de visión artificial, con base en el análisis de las fotos del entorno local tomadas por las cámaras del manipulador redundante.

Para calcular el movimiento adecuado, se generará, en primer lugar, la trayectoria cartesiana de una de las manos del robot, con base en la ubicación del objeto, y, luego, se calculará la cinemática inversa del manipulador redundante para esta trayectoria cartesiana.

Ahora bien, antes de calcular las trayectorias, se determina el espacio de trabajo del manipulador, a través de:

1) Cinemática directa, por medio de parámetros de Denavit-Hartenberg (D-H), de todo el manipulador (ver tablas 2 y 3. Se hace igual para los miembros izquierdos y los derechos). Se denota que, para los brazos, se aplican los parámetros D-H modificados, debido a la configuración cinemática que tienen.

2) Se hacen barridos de las articulaciones, desde sus ángulos mínimos hasta máximos. Con esto, es posible definir los límites de movimiento de la mano del robot (*end-effector*).

Una vez se haya definido el espacio de trabajo (ET), viable crear trayectorias cartesianas, el *end-effector*, dentro de este ET, pues es fundamental que el movimiento de la mano esté dentro del alcance del robot. Para calcular el ET se utilizan las fórmulas obtenidas en el vector de traslación de la matriz de transformación homogénea, que relaciona el sistema de referencia del *end-effector* con el sistema de base del brazo respectivo (vea ecuaciones 1 a 3).

Tabla 2. Parámetros de Denavit-Hartenberg modificados para brazo izquierdo.

i	α (Alpha)	a	θ (Theta)	d (distance)
1	$-\pi/2$	0	q1	0
2	$\pi/2$	0	q2	0
3	$-\pi/2$	0	q3	d3
4	$\pi/2$	0	q4	0
5	$-\pi/2$	0	q5	d5

Fuente: elaboración propia.

Tabla 3. Parámetros de Denavit-Hartenberg modificados para pierna izquierda.

i	α (Alpha)	a	θ (Theta)	d (distance)
1	$-3\pi/4$	0	q1	0
2	$-\pi/2$	0	q2	0
3	$\pi/2$	0	q3	0
4	0	a4	q4	0
5	0	a5	q5	0
6	$-\pi/2$	0	q6	0

Fuente: elaboración propia.

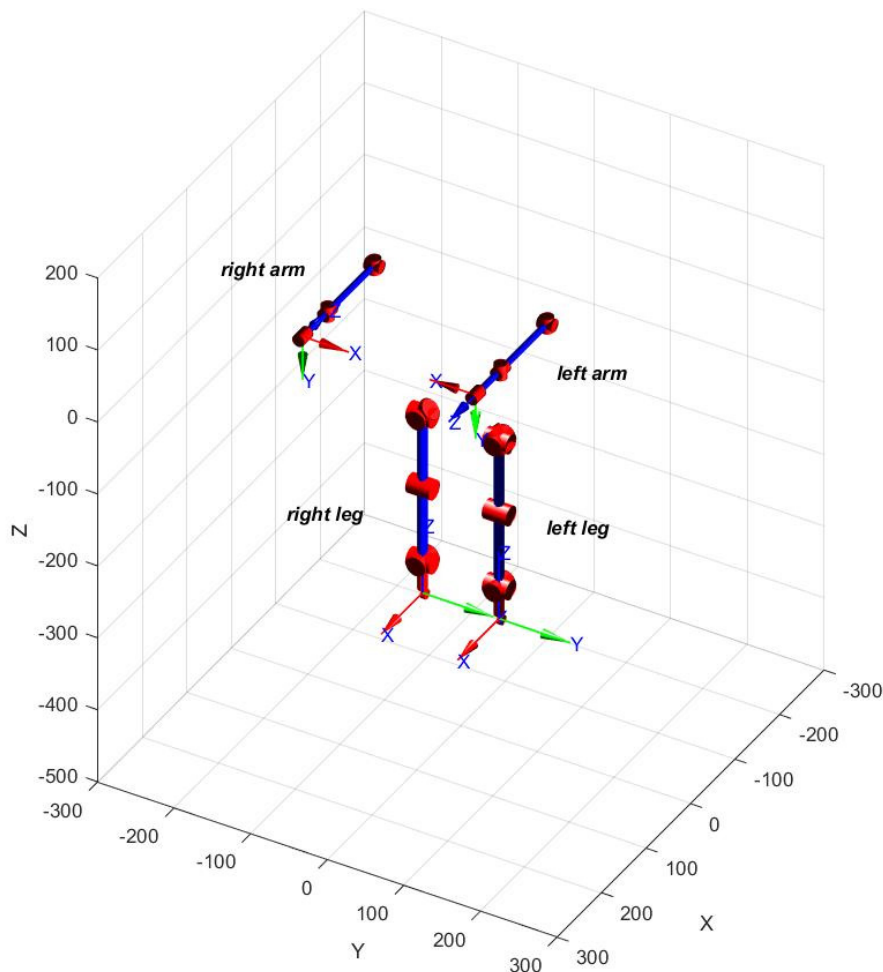
$$x = (d4 * \sin(q1) * \sin(q3) * \sin(q4) - \cos(q1) * (\sin(q2 - \pi/2) * ((d4 * \cos(q4)) + l2) + (d4 * \cos(q3) * \cos(q2 - \pi/2) * \sin(q4)))) \quad (1)$$

$$y = \cos(q2 - \pi/2) * ((d4 * \cos(q4)) + d2) - (d4 * \cos(q3) * \sin(q4) * \sin(q2 - \pi/2)) \quad (2)$$

$$z = \sin(q1) * (\sin(q2 - \pi/2) * ((d4 * \cos(q4)) + d2) + (d4 * \cos(q3) * \cos(q2 - \pi/2) * \sin(q4))) + (d4 * \cos(q1) * \sin(q3) * \sin(q4)) \quad (3)$$

El modelo de la parte manipuladora, a partir de los parámetros D-H (Corke, 2007), se puede animar según se muestra en la figura 3.

Figura 3. Resultado de animación de robot manipulador redundante (humanoide), aplicando parámetros de Denavit-Hartenberg.



Fuente: elaboración propia.

Navegación

En cuanto a la navegación, se emplea el método del Lugar de las raíces para generar una trayectoria que permita evadir las colisiones con los obstáculos localizados. Recordemos que este

método aprovecha la propiedad de las fuerzas de atracción (entre polo y cero) y repulsión (entre cero y cero, o polo y polo).

Figura 4. Imagen del laboratorio CUN en la que se visualizan el robot asistencial, los obstáculos simulados (rectángulos azules), la posición objetivo (rectángulo rojo) y el resto del mobiliario.



Fuente: elaboración propia.

En esta parte se realizan los siguientes pasos:

1) Asumir que la zona de trabajo o de desplazamiento del robot asistencial (ver figura 4.) es el plano real-complejo, donde se añaden los polos y ceros.

2) Colocar un cero en el punto de inicio de la trayectoria donde inicia la navegación, es decir, en el centro del robot.

3) Colocar un polo en el punto objetivo hacia donde se necesita que se mueva el robot.

4) Colocar una pareja polo-cero en cada obstáculo en el camino. Cada una de estas parejas va a ejercer fuerzas de repulsión cuando la trayectoria de navegación pase cerca del obstáculo, deformándola. Por lo tanto, esta trayectoria evita que el robot colisione con los obstáculos.

La síntesis de la aplicación de esta metodología se detalla en las ecuaciones 4 a 9 (Martínez,

Orejuela y Arbulú, 2018; Arbulú, Martínez y Montiel, 2015):

$$H_1(s) = \frac{(s + 1 - 4i)(s + 1 + 4i)}{(s + 14 - 6i)(s + 14 + 6i)} \quad (4)$$

$$H_2(s) = \frac{(s + 10.7 - 5i)(s + 10.7 + 5i)}{(s + 11.3 - 5i)(s + 11.3 + 5i)} \quad (5)$$

$$H_3(s) = \frac{(s + 6.7 - 8i)(s + 6.7 + 8i)}{(s + 7.3 - 8i)(s + 7.3 + 8i)} \quad (6)$$

$$H_4(s) = H_1(s) * H_2(s) * H_3(s) \quad (7)$$

$$H_5(s) = \frac{(s + z_r + z_i * i)(s + z_r - z_i * i)}{(s + p_r + p_i * i)(s + p_r - p_i * i)} \quad (8)$$

$$H_6(s) = H_4(s) * H_5(s) \quad (9)$$

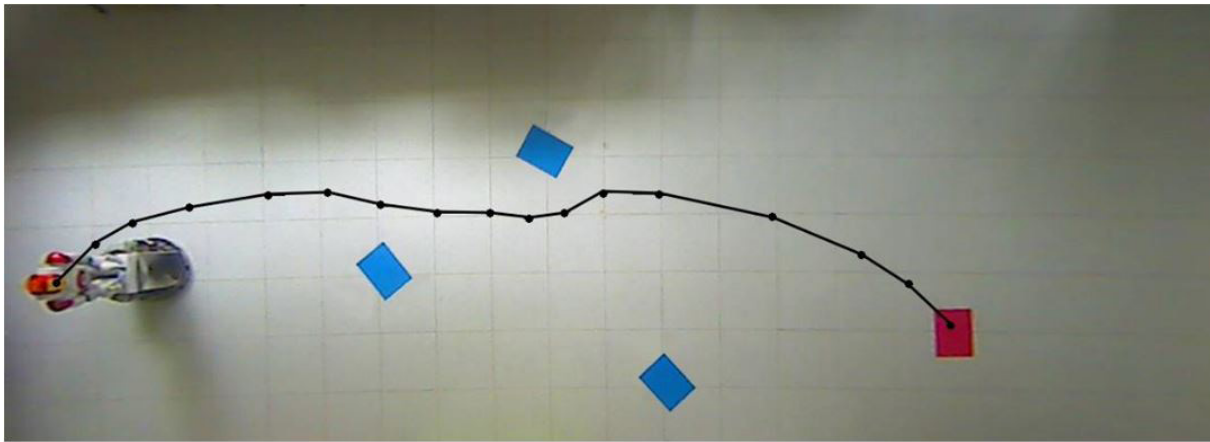
Resultados

En esta sección se presentan los resultados obtenidos al aplicar los algoritmos descritos en el apartado anterior.

En la figura 5 se ve superpuesta y discretizada la trayectoria de navegación calculada, dado que, en los cálculos teóricos, se tiene como resultado una trayectoria continua; entonces, para

implementarla en la práctica, se debe discretizar (los puntos sobre la trayectoria obtenida serán los puntos de interés, que se introducirán en el cálculo del movimiento de la plataforma omnidireccional). Con esto, se ha logrado con éxito el desplazamiento autónomo del robot en una oficina, evitando chocar con los obstáculos simulados (rectángulos azules), y llegando a la posición objetivo (rectángulo rojo).

Figura 5. Trayectoria de navegación del robot para llegar a la posición objetivo, evadiendo los obstáculos.

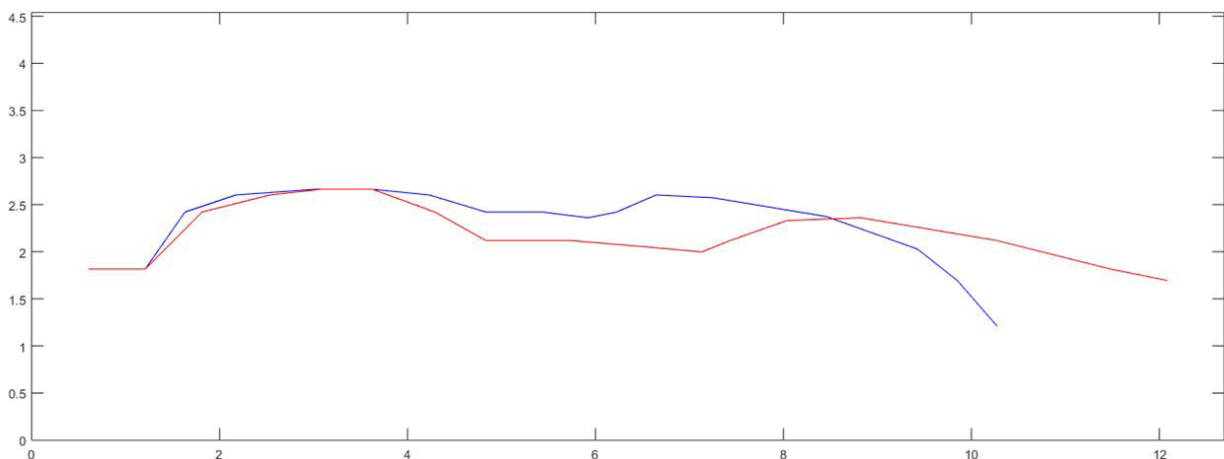


Fuente: elaboración propia.

Sin embargo, si bien se cumplió el objetivo de evadir los obstáculos, se encuentra una diferencia entre el recorrido real del robot en relación con el teórico (ver figura 6.). Esto se debe al deslizamiento omnidireccional del

robot omnidireccional, en superficie de desplazamiento, y a errores en control de posición de las ruedas. Con un controlador global se podría mejorar el desplazamiento, según la trayectoria requerida.

Figura 6. Trayectoria de navegación del robot calculada por algoritmo de Lugar de raíces (azul), superpuesta con la trayectoria que realmente hace el robot (rojo).

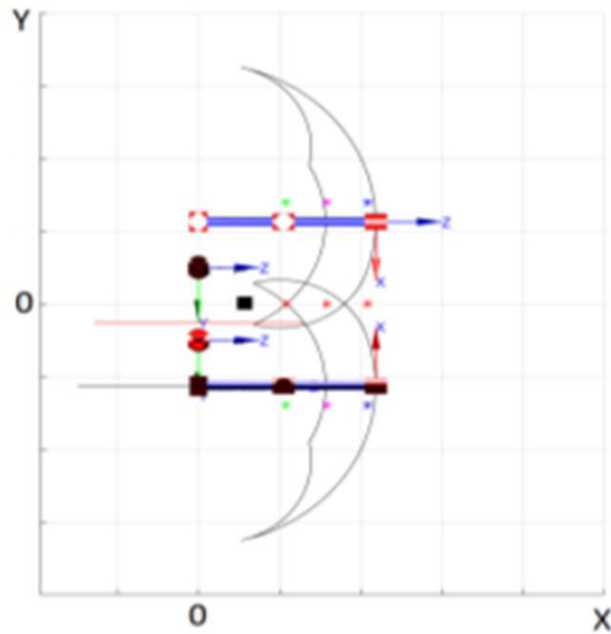


Fuente: elaboración propia.

Para efectos de manipulación, se han calculado los límites del espacio de trabajo del manipulador redundante (ver figuras 7 y 8); incluso, también el espacio de trabajo aumentado (figura 8,

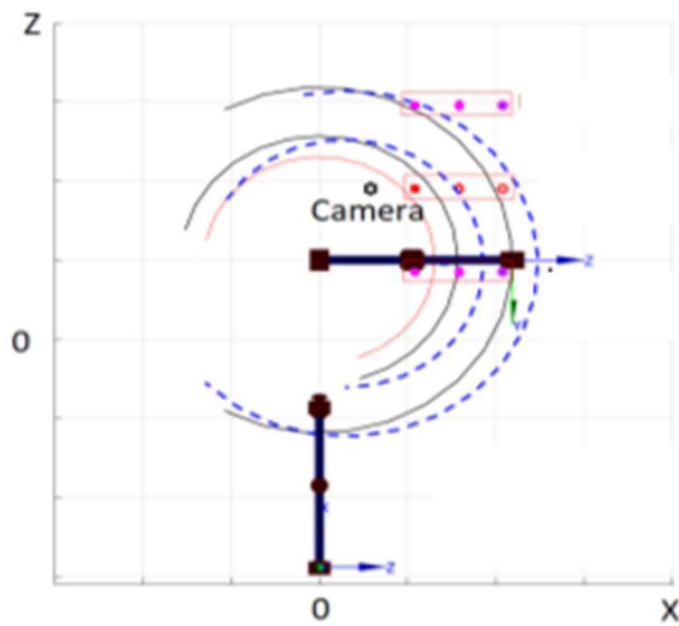
línea azul punteada), puesto que se dispone de un torso que puede girar hacia adelante y hacia atrás, o de izquierda a derecha.

Figura 7. Espacio de trabajo del manipulador redundante - vista superior (la línea azul punteada es el espacio de trabajo aumentado, debido a giro de torso).



Fuente: elaboración propia.

Figura 8. Espacio de trabajo del manipulador redundante - vista lateral (la línea azul punteada es el espacio de trabajo aumentado, debido a giro de torso).

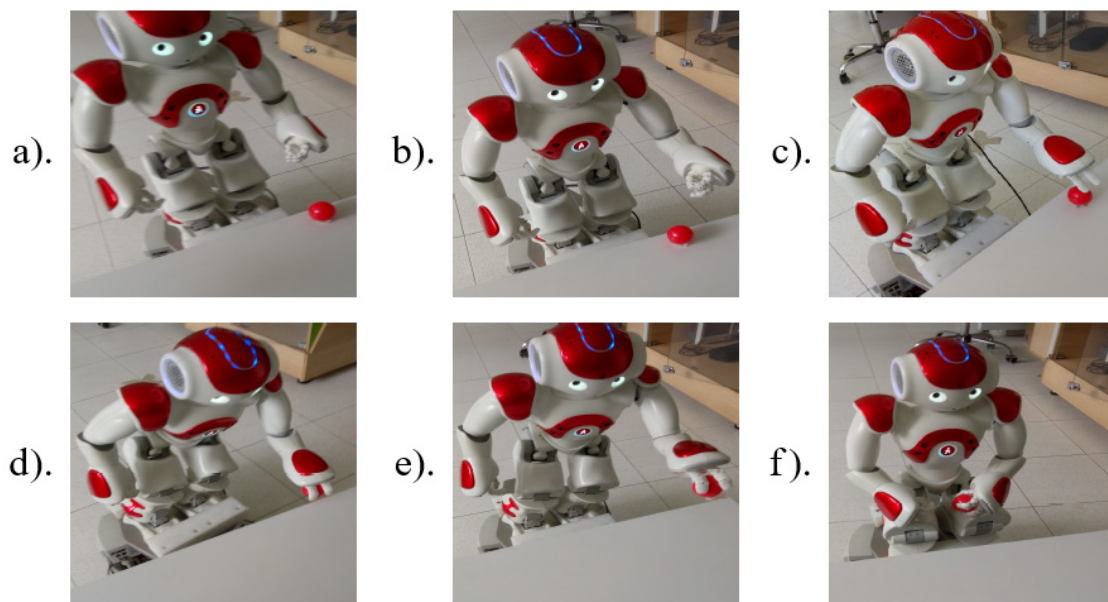


Fuente: elaboración propia.

Con el espacio de trabajo delimitado, se propusieron trayectorias que están incluidas en esta zona. Por lo tanto, será posible planificar el movimiento de recogida y despacho de objetos. Tal como se muestra en la figura 9, se cumple el objetivo de coger un objeto satisfactoriamente, con índices mínimos de error (en torno al 97 %

de efectividad de certeza). Este indicador fue medido luego de numerosas pruebas experimentales. Ahora bien, en posteriores trabajos se plantea involucrar algoritmos de inteligencia artificial para mejorar la autonomía y respuesta del robot (Martínez, Rendón, y Arbulú, 2018).

Figura 9. Prueba experimental de manipulación: robot asistencial recogiendo un objeto de manera autónoma, con asistencia de visión local de robot manipulador redundante.



Fuente: elaboración propia.

Conclusiones

La primera conclusión es que se cumple el objetivo de desarrollar subtareas de locomoción y manipulación autónomas, dado que se demuestra, en la experimentación, que es viable aplicar estos algoritmos. En segundo lugar, se detecta que, en el algoritmo de navegación, en relación con la localización de polo-cero en los obstáculos, debe realizarse la diagonal del rectángulo que encierra el objeto que simula el obstáculo; de esta manera, se tiene la repulsión suficiente para evitar colisiones.

Por otro lado, se logra implementar el espacio de trabajo del manipulador redundante para delimitar, previamente, su movimiento cartesiano y alcance. Con base en estas conclusiones, se están desarrollando trabajos que se centran en la integración de locomoción con manipulación para desarrollar tareas autónomas. Posteriormente, se implementarán algoritmos de inteligencia artificial para alcanzar la autonomía absoluta de los robots.



Referencias

- Arbulú, M., Martínez, F. y Montiel, H. (2015). Metodología para el uso de la técnica de localización de raíces en la planeación de rutas para robots móviles. *Tecnura*, 19(46), 49-64. DOI: <https://doi.org/10.14483/udistrital.jour.tecnura.2015.4.a04>
- Corke, P. (2007). A simple and systematic approach to assigning Denavit-Hartenberg parameters. *IEEE Transactions on Robotics*, 23(3), 590-594. Recuperado de <https://bit.ly/3hEMhxA>
- Hayne, R., Luo, R. y Berenson, D. (2016). Considering avoidance and consistency in motion planning for human-robot manipulation in a shared workspace. *IEEE International Conference on Robotics and Automation (ICRA)*, 3948-3954. Recuperado de <https://bit.ly/3wHT4Le>
- Kemp, C., Edsinger, A. y Torres, E. (2007). Challenges for robot manipulation in human environments [Grand Challenges of Robotics]. *IEEE Robotics and Automation Magazine*, 14(1), 20-29. Recuperado de <https://ieeexplore.ieee.org/document/4141029>
- Kruse, D., Radke, R., Wen, J. (2015). Collaborative human-robot manipulation of highly deformable materials. *IEEE International Conference on Robotics and Automation (ICRA)*, 3782-3787. Recuperado de <https://bit.ly/3ep8ZI5>
- Martínez, F., Orjuela, S. y Arbulú, M. (2017). Global navigation approach for assistant robot. *Tecnura*, 21(51), 105-117. DOI: <https://doi.org/10.14483/udistrital.jour.tecnura.2017.1.a08>
- Martinez, F., Rendón, A., y Arbulú, M. (2018). A data-driven path planner for small autonomous robots using deep regression models. En Y. Tan, Y. Shi y Q. Tang (eds.), *Data Mining and Big Data. DMBD 2018. Lecture Notes in Computer Science*, 10 943. Springer: Cham. Recuperado de <https://bit.ly/3wCWBKQ>