

Hacia simulaciones realistas de agarre robótico submarino en el contexto del proyecto TWINBOT

Mario Ávila, Alejandro Solis, Raúl Marín-Prades, Pedro J. Sanz, José V. Martí
 Universitat Jaume I de Castellón
 al409457, asolis, rmarin, sanzpj, vmarti@uji.es

Resumen

El uso de tecnologías de videojuegos en las aplicaciones robóticas está siendo de gran ayuda tanto en el diseño de interfaces de usuario como en la implementación de servidores de simulación, que permitan un mayor realismo en las imágenes generadas por computador. Así mismo, cuando se procura realizar una simulación donde existe contacto de un brazo manipulador, como por ejemplo en tareas de peg-in-hole, estas tecnologías no están todavía suficientemente avanzadas. En este artículo describimos el estado actual de uno de los experimentos de simulación del proyecto TWINBOT, donde se procura realizar tareas de manipulación con tuberías, incluyendo recuperación, transporte e inserción. El sistema utiliza la nueva versión de la física de Unity, a través del objeto Articulation Body, el cual da un mayor grado de precisión en el cálculo de fuerzas tras una interacción robótica con un objeto. El escenario de la simulación es submarino, pudiendo apreciar el nivel actual alcanzado con este tipo de sistemas.

Palabras clave: Simulaciones, Robótica Submarina, Manipulación, Interacción

1 INTRODUCCIÓN

El proyecto TWINBOT [5] tiene como uno de sus objetivos la exploración de nuevas técnicas de agarre, transporte y ensamblado cooperativo de tuberías con vehículos submarinos autónomos supervisados manipuladores. Para llevar a cabo los experimentos que den pie al cumplimiento de los objetivos planteados en el proyecto se dispone de dos vehículos G500 [4] y dos brazos manipuladores Reach Bravo 7 de Blueprint Robotics. Se trata de experimentos complejos, donde poner en marcha cada uno de los robots G500 de forma cooperativa conlleva un gran esfuerzo científico y humano.

El contar con un entorno de simulación es de gran importancia hoy en día en el ámbito de la robótica ya que puede ayudar a reducir la complejidad de los problemas que uno puede encontrar al hacer un experimento en un escenario real. Realizando

pruebas simuladas antes de pasar a un experimento real se pueden mitigar pérdidas económicas y de tiempo. En el campo de la robótica submarina existen algunos ejemplos de entornos de simulación. Uno de ellos, UWSim[3], es una herramienta de software abierto que sirve para simular y visualizar misiones de intervención submarinas. Otro ejemplo es el UUV Simulator[2], el cual es una extensión de Gazebo para escenarios subacuáticos que permite la simulación de múltiples vehículos submarinos y tareas de intervención utilizando manipuladores robóticos.

Recientemente los entornos de simulación actuales han ido avanzando enormemente gracias al esfuerzo de la industria de los videojuegos. Aún así, esta tecnología todavía no estaba pensando en potenciales aplicaciones de interacción y simulación con sistemas físicos. Poco a poco estas empresas han empezado a trabajar en esta línea, permitiendo entornos de simulación realista que puedan dotar de datos al aprendizaje de comportamientos autónomos, especialmente en la línea de la automoción.

A modo de ejemplo, Unity ha ofrecido el repositorio de código abierto "Unity Robotics Hub", el cual incluye la integración con ROS (Robot Operating System), importación de ficheros URDF, y su integración con la nueva versión de la física interna, la cual ya proporciona un objeto especialmente diseñado para la simulación de brazos robóticos, el ArticulationBody.

El propósito de esta publicación es mostrar la manera en que se puede utilizar la plataforma para desarrollo de videojuegos Unity para crear un entorno de simulación que pueda complementar/apoyar las tareas que se han llevado a cabo en el proyecto TWINBOT.

En la figura 1 se puede apreciar una escena simulada del escenario inicial del experimento.

2 ARQUITECTURA

El sistema que estamos presentando en este artículo sigue la arquitectura mostrada en la figura 2. Como podemos apreciar el sistema de simu-

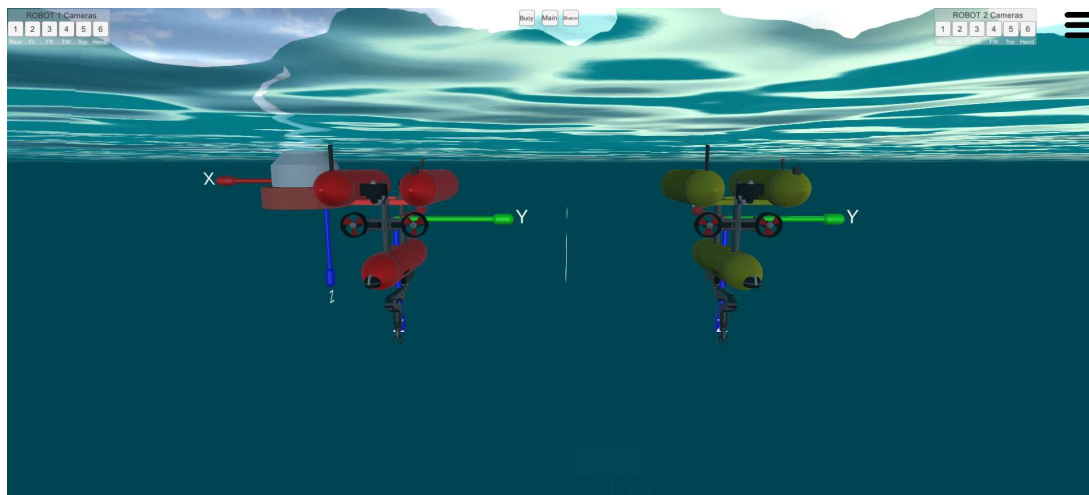


Figura 1: Escenario inicial de simulación en el ámbito de la robótica cooperativa submarina (proyecto TWINBOT)

lación ofrece servicios via TCP, tanto a través de TCP como UDP.

Esta interfaz de comunicación permite la interacción con los scripts de la interfaz de usuario, que permite interactuar con el sistema de forma supervisada, recibiendo las imágenes de las cámaras, y creando nuevas trayectorias.

Precisamente, la planificación de las trayectorias es realizada a través del sistema MoveIt de ROS. MoveIt recibe la posición de las articulaciones de los brazos robóticos y calcula las nuevas posiciones articulares para poder ejecutar la trayectoria que lleve al efector final a la posición deseada. Finalmente, el nodo de la interfaz se encarga de enviar estas nuevas posiciones a los brazos a través de comandos TCP.

3 INTERFAZ DE USUARIO

La interfaz de usuario del sistema que se presenta en este artículo está implementada con Python, usando la librería de widgets gráficos TKinter, e interactuando con el simulador y planificador de trayectorias MoveIt a través del sistema ROS y TCP/IP.

Como podemos apreciar en la figura 3, el sistema permite monitorizar el estado de las cámaras remotas, controlar el robot de forma manual, y establecer las coordenadas del efector final del manipulador.

La supervisión y control del planificador MoveIt puede realizarse también haciendo uso del visualizador ROS Rviz, tal cual se puede apreciar en la figura 4. Este permite establecer de forma manual la posición deseada del efector final o de las distintas articulaciones del robot y visualizar la

trayectoria calculada antes de que esta sea ejecutada.

4 SIMULACIÓN

En este apartado se presentan los pasos realizados por el sistema robótico semi-autónomo, desde la posición inicial, los primeros movimientos del robot manipulador, la aproximación, y el agarre final.

Para llevar a cabo el agarre de la tubería es necesario realizar un aproximamiento inicial que sitúe al robot móvil sobre el centro de gravedad de la misma. Para ello se ha implementado un servocontrol visual encargado de realizar una primera aproximación a la tubería. Con el fin de realizar este control, es necesario extraer la tubería de la imagen y tras esto, obtener características necesarias para el control.

Dado que es conocido el color de la tubería, a partir de la imagen capturada por la cámara del manipulador, se extrae el canal de saturación de esta, ya que es el que presenta un mayor contraste entre el objeto y el fondo de la escena (imagen 5 b)). Tras esto se realiza una umbralización del canal S y se descartan todas las regiones buscando al de mayor excentricidad (5 c)). De este modo se eliminan las regiones parasitas asegurando que el único objeto identificado sea la tubería. A pesar del ruido introducido en la imagen por la simulación del entorno marino, este algoritmo permite extraer la tubería a una distancia de hasta 7,5 metros de altura. Para asegurar la facilidad de uso, la interfaz gráfica resalta con un rectángulo rojo la tubería una vez detectada. Es responsabilidad del operario manipular al robot de forma manual hasta que se encuentre a distancia sufi-

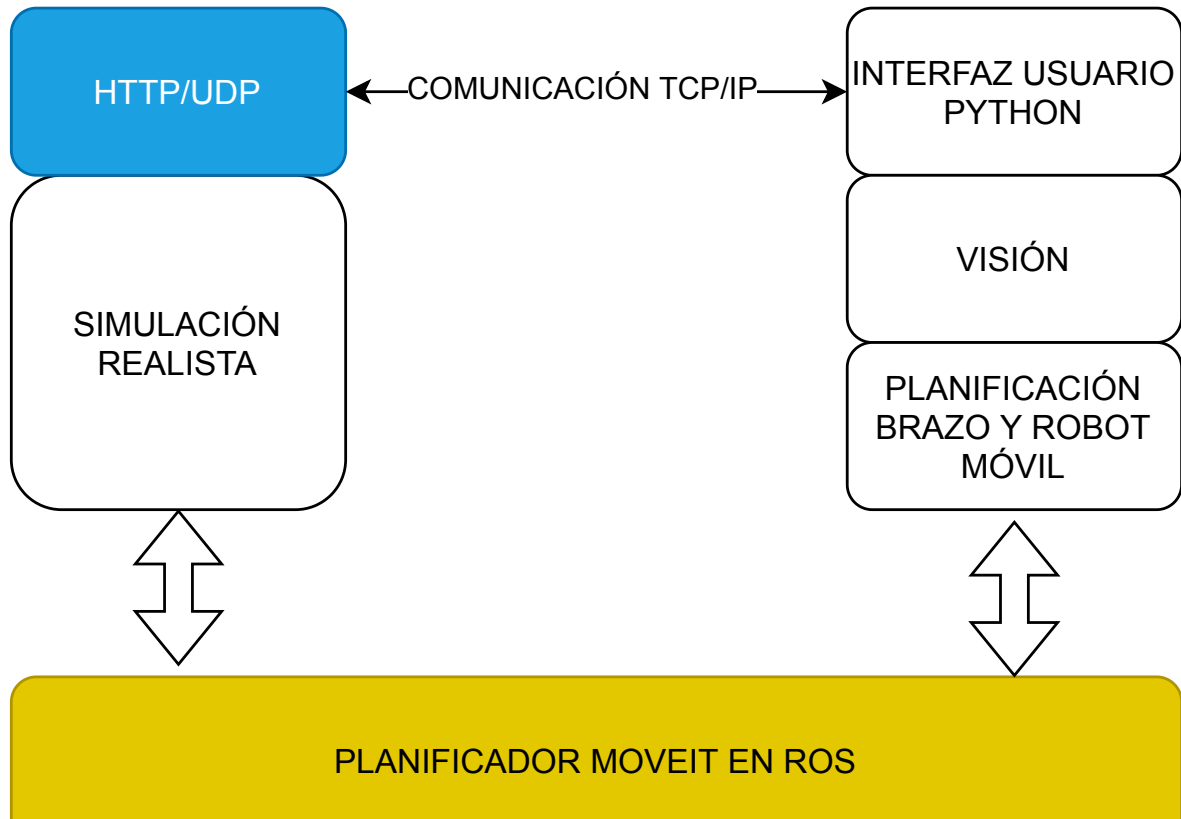


Figura 2: Arquitectura del Experimento de simulación

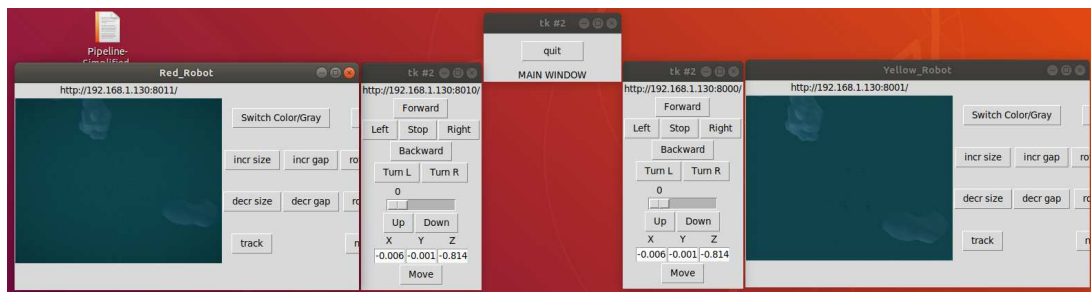


Figura 3: Interfaz de control de la planificación del robot manipulador móvil y la supervisión del estado de las cámaras

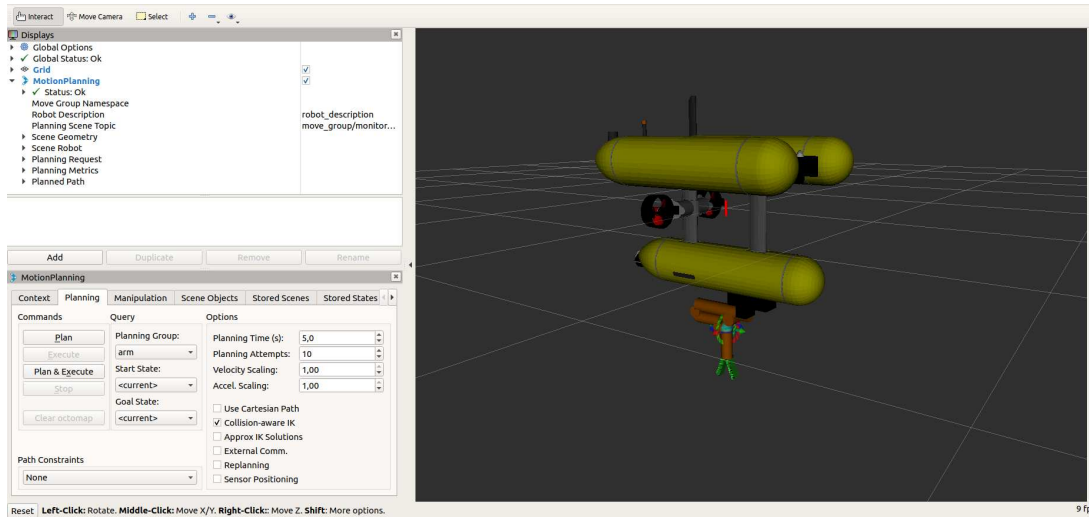


Figura 4: Escenario inicial de simulación en el ámbito de la robótica cooperativa submarina (proyecto TWINBOT)

ciente como para detectar la tubería.

Conocidas las dimensiones y el color de la tubería, puede extraerse de la imagen las características necesarias para el control tal y como se muestra en la imagen 5 d). A través de librerías de visión artificial como OpenCv, puede extraerse las coordenadas del centroide de esta. Este centroide se utilizará como aproximación del centro de masas del objeto y como punto de referencia para la tarea de servocontrol. A partir del primer autovector del eje de mínima inercia de la tubería, puede obtenerse el ángulo que forma la tubería con el eje x de la cámara.

La última característica necesaria para realizar el control por visión, es la distancia Z entre la cámara y el objeto. Si bien no es posible obtener la posición exacta del objeto con una única cámara, conociendo los parámetros intrínsecos de esta y las dimensiones del objeto, puede estimarse la distancia por triangulación, tal y como se explica en la ecuación 1.

$$Z(mm) = \frac{f(mm) * rl(mm) * height(pxls)}{ol(pxls) * sensheight(mm)} \quad (1)$$

Siendo rl el largo real del objeto, im la altura de la imagen en píxeles, ol la longitud del objeto en píxeles de la imagen y sens height la altura del sensor de la cámara.

Para el control de aproximación a la tubería se ha decidido realizar un servocontrol visual 2-1/2D ya que tal y como se explica en [1], esta forma de servocontrol permite desacoplar el control de posición lineal del control de posición angular. Además, este algoritmo soluciona las principales desventa-

jas de las técnicas de servocontrol 2D y 3D, presentando un mayor ratio de convergencia que los controles basados en la imagen y una menor posibilidad de perder el objetivo que los controladores espaciales. La librería Visp facilita el diseño de la ley de control, siendo necesario únicamente extraer la posición del objeto relativo a la cámara en el plano imagen, la orientación y la distancia del objeto a la misma, y especificar la posición final deseada. Si bien el sistema de control calcula la velocidades necesarias para alcanzar la tubería respecto al sistema de referencia de la cámara. Es conocida la relación entre la cámara y el centro de gravedad de los Girona500 en cada momento, por lo que puede aplicarse un cambio en el sistema de referencia (ecuación 2).

$$\dot{r} = \begin{bmatrix} {}^G V \\ {}^G W \end{bmatrix} = \begin{bmatrix} {}^G R_c V_c - {}^G R_c W_c \times {}^G T_c \\ {}^G R_c W_c \end{bmatrix} \quad (2)$$

De esta forma, se obtienen las velocidades lineales y angulares necesarias para mover la cámara del robot a la posición deseada respecto al objeto.

Debido a las dimensiones del objeto, el servocontrol visual no puede aproximar al Girona 500 a menos de 2 metros de distancia de este, ya que no se podría captar la tubería en su totalidad en la imagen y se perdería la referencia. Por ello, una vez situado perpendicularmente sobre la tubería una corta distancia, se desactiva el servocontrol y se realiza un desplazamiento descendente con el Girona500 hasta que la tubería entre en el espacio de trabajo efectivo del robot. Esto se comprueba a través de un sensor laser situado en el efector final. Al realizarse el movimiento en una distancia tan corta (de menos de 2 metros)

se supone que el robot no se desviará lo suficiente del punto de referencia como para imposibilitar el agarre.

Una vez en el espacio de trabajo efectivo del manipulador, se realiza un movimiento del efector final hacia el objeto y se realiza el agarre de la tubería.

Agradecimientos

Este trabajo ha sido realizado por el equipo del CIRTESU, "Centro de Investigación en Robótica y Tecnologías Subacuáticas" de la Universidad Jaume I, y financiado por los proyectos DPI2017-86372-C3 (TWINBOT), IDIFEDER/2018/013 (GV), UJI-B2018-34 (NEPTUNO), PID2020-115332RB-C31 (COOPERAMOS), y el proyecto H2020-Peacetolero-NFRP-2019-2020-04.

English summary

TOWARDS REALISTIC SIMULATIONS OF UNDERWATER ROBOTIC GRASPINGS IN THE CONTEXT OF THE TWINBOT PROJECT

Abstract

The use of video game technologies in robotic applications is being of great help both in the design of user interfaces and in the implementation of simulation servers, which allow greater realism in computer-generated images. Likewise, when trying to perform a simulation where there is contact of a manipulator arm, such as in peg-in-hole tasks, these technologies are not yet sufficiently advanced. In this article we describe the current state of one of the simulation experiments of the TWINBOT project, which attempts to carry out handling tasks with pipes, including recovery, transport and insertion. The system uses the new version of Unity physics, through the Articulation Body object, which gives a greater degree of precision in the calculation of forces after a robotic interaction with an object. The simulation scenario is underwater, being able to appreciate the current level reached with this type of system.

Keywords: Simulations, Marine

Robotics, Manipulation, Interaction

Referencias

- [1] E. Malis, F. Chaumette, and S. Boudet. 2 1/2 d visual servoing. *IEEE Transactions on Robotics and Automation*, 15(2):238–250, 1999.
- [2] M. M. M. Manhães, S. A. Scherer, M. Voss, L. R. Douat, and T. Rauschenbach. Uuv simulator: A gazebo-based package for underwater intervention and multi-robot simulation. In *OCEANS 2016 MTS/IEEE Monterey*, pages 1–8, 2016.
- [3] M. Prats, J. Pérez, J. J. Fernández, and P. J. Sanz. An open source tool for simulation and supervision of underwater intervention missions. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2577–2582, 2012.
- [4] D. Ribas, P. Ridao, L. Magí, N. Palomeras, and M. Carreras. The girona 500, a multi-purpose autonomous underwater vehicle. In *OCEANS 2011 IEEE - Spain*, pages 1–5, 2011.
- [5] R. Pi, P. Cieslak, P. Ridao, and P. J. Sanz. Twinbot: Autonomous underwater cooperative transportation. *IEEE Access*, 9:37668–37684, 2021.



© 2021 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution CC BY-NC-SA 4.0 license (<https://creativecommons.org/licenses/by-nc-sa/4.0/deed.es>).

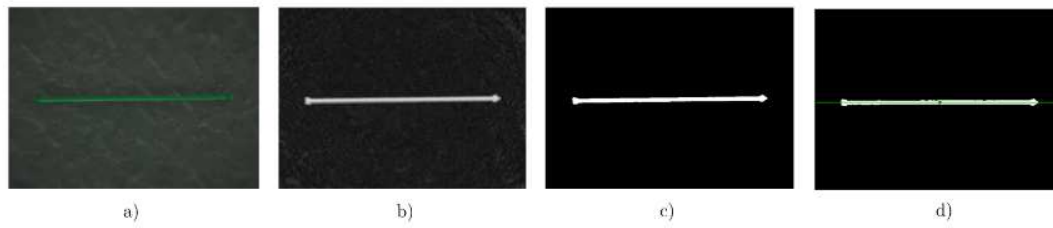


Figura 5: Proceso de segmentación y extracción de características de la tubería. a) Imagen original; b) Canal de saturación; c) Binarización y eliminación de regiones parasitas; d) obtención del centroide y el eje de mínima inercia

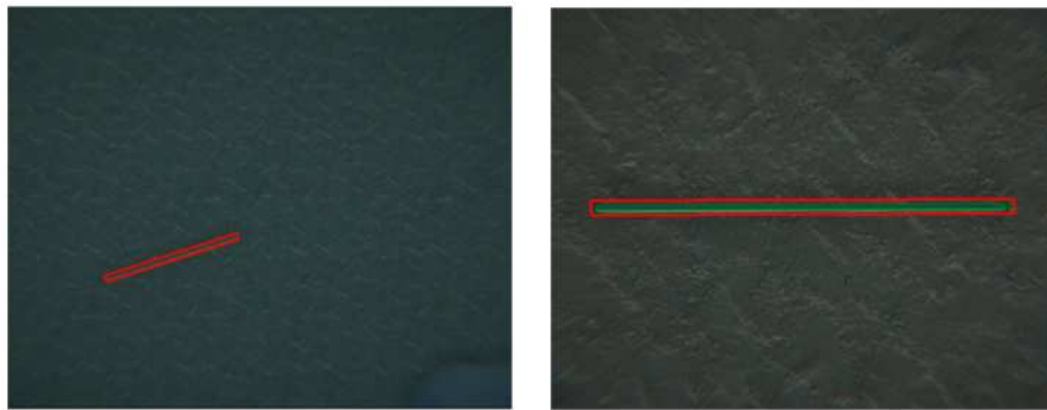


Figura 6: (Izquierda) Posición inicial del control visual. (Derecha) Posición final del control visual.

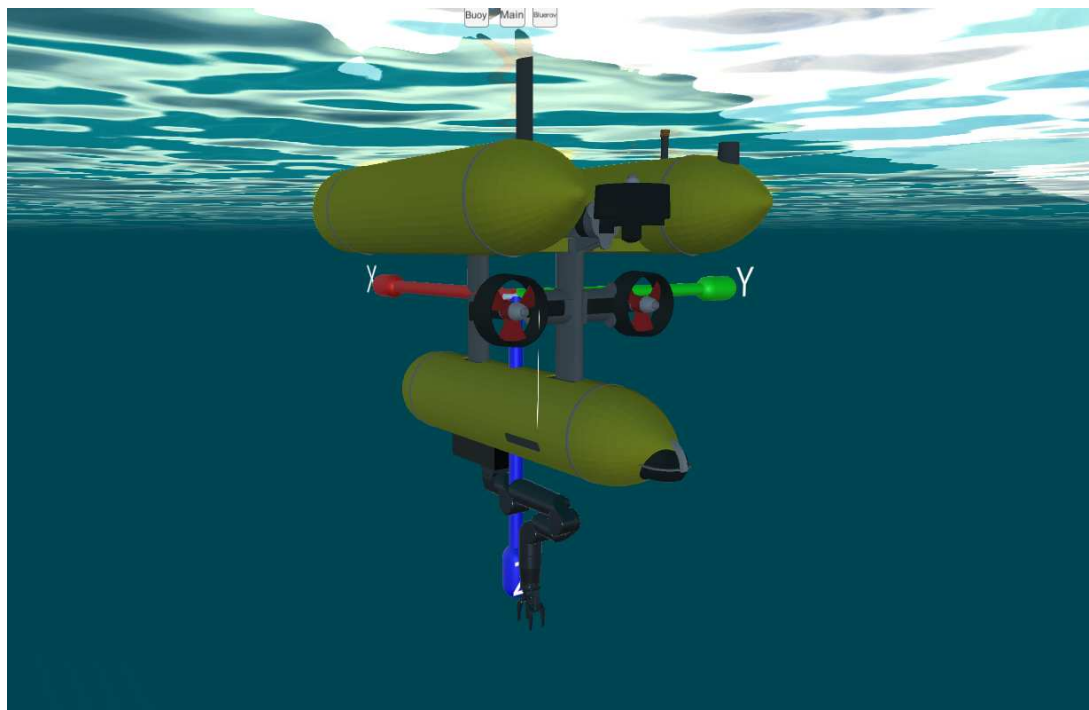


Figura 7: Posición inicial del robot como preparación del experimento de agarre de la tubería

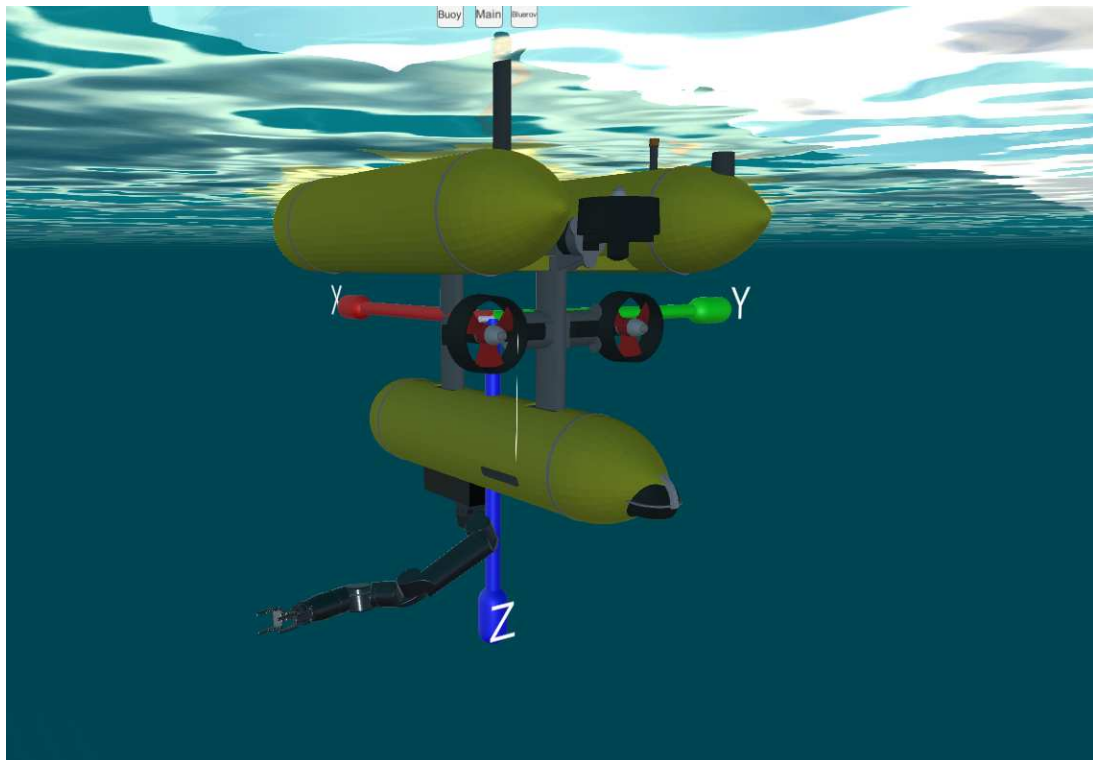


Figura 8: Primera prueba de planificación de movimiento del brazo manipulador



Figura 9: Aproximación a la tubería



Figura 10: Contacto con la tubería: Vista Superior



Figura 11: Contacto con la tubería: Vista Frontal



Figura 12: Agarre realizado correctamente