

APRENDIZAJE BASADO EN PROYECTOS COMO METODOLOGÍA PARA UN ACERCAMIENTO A SLAM

David Gallarta-Sáenz, Javier Rico-Azagra, Montserrat Gil-Martínez

{david.gallarta, javier.rico, montse.gil}@unirioja.es

Grupo de Investigación de Ingeniería de Control, Departamento de Ingeniería Eléctrica
Universidad de La Rioja

Resumen

Este trabajo presenta una primera toma de contacto para el alumnado universitario con la técnica SLAM (Simultaneous Localization And Mapping) a través de la metodología de Aprendizaje Basado en Proyectos (ABP). El SLAM es una técnica cuyo objetivo es dotar a un robot móvil de la capacidad de localización y posicionamiento en un entorno desconocido, generando el mapa mientras navega por él. También se muestra el enfoque docente dado al tema dentro de una asignatura, así como algunos conceptos teóricos básicos a partir de los cuales ir desarrollando un proyecto sencillo. En cuanto al proyecto en sí, la aplicación implementada ejecuta SLAM utilizando la Navigation Toolbox y la ROS Toolbox de MATLAB junto con ROS (Robot Operating System), un entorno de trabajo específico para robots, que se instala en una Raspberry Pi 3. Con ella se dota de autonomía a dos plataformas móviles terrestres que incorporan un sensor láser de tipo lidar (Laser Imaging Detection And Ranging) para obtener los datos del entorno.

Palabras clave: ABP, SLAM, Robot móvil, MATLAB, ROS, Lidar.

1. INTRODUCCIÓN

En el marco del Grado en Ingeniería Electrónica Industrial y Automática impartido en la Escuela Técnica Superior de Ingeniería Industrial de la Universidad de la Rioja, una de las asignaturas optativas del 4º curso es Sistemas robotizados y, según el plan de estudios, su temario es:

- Tema 1: Introducción a la robótica móvil
- Tema 2: Características cinemáticas y modelado de robots móviles
- Tema 3: Control de robots móviles
- Tema 4: Generación de trayectorias para robots móviles
- Tema 5: Navegación en robots móviles
- Tema 6: Localización de robots móviles

En este sentido, como proyecto final de la asignatura, se plantea la posibilidad de incluir un campo muy en boga de la comunidad investigadora en robótica móvil, así como en el ámbito industrial debido a su versatilidad, a través del Aprendizaje Basado en Proyectos. Este contenido es SLAM, un tema que da la opción de combinar de forma práctica muchos de los conceptos impartidos y desarrollarlo mediante la citada metodología. El SLAM [5, 16] es una técnica relevante ya que soluciona el problema de posicionamiento interior. Sin embargo, los errores acumulados por los sensores, la aproximación en los modelos y la gran cantidad de información procesada la convierten en una tarea compleja.

La robótica móvil [18, 20] es una rama de la robótica centrada en plataformas móviles, generalmente autónomas, capaces de desplazarse en entornos parcial o totalmente desconocidos. Para que estos robots realicen sus operaciones de forma correcta, deben reunir cuatro requisitos: control, percepción, localización y planificación.

En este punto se plantea el principal reto relacionado con la robótica móvil: para localizar el robot necesitamos un mapa, pero, para generar el mapa se necesita conocer la localización del robot. La localización [3] es un problema al que se ha enfrentado el ser humano a lo largo de la historia, por ejemplo, los primeros marineros creaban sus mapas a medida que navegaban, utilizando características naturales que iban observando. Y, de forma análoga, los robots móviles actuales que no pueden utilizar GPS se enfrentan al mismo problema [19].

Una vez introducidos los aspectos técnicos básicos, los objetivos del trabajo, con los cuales afianzar y completar los conocimientos de la asignatura de un modo más práctico, son los siguientes:

- Verificar la utilidad del ABP en la docencia de ingeniería a nivel universitario .
- Poner en funcionamiento un lidar como sensor para la recogida de datos a partir de los cuales crear los mapas de espacios interiores.
- Implementar SLAM empleando MATLAB y ROS, junto con los datos del lidar.

En la Figura 1 se presenta el esquema de la arquitectura del sistema, los dispositivos *hardware* empleados y el *topic* (/scan) de ROS en el que se publican los datos para implementar el SLAM.

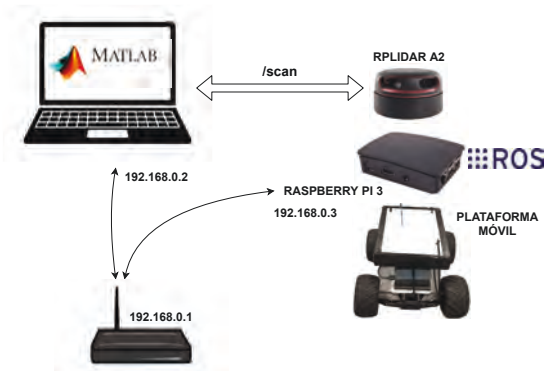


Figura 1: Arquitectura del sistema.

El presente trabajo se organiza en seis secciones: en la segunda se describe el contexto académico; la tercera presenta el sistema y las herramientas empleadas; la cuarta explica el desarrollo técnico; en la quinta sección se presentan los resultados obtenidos y, por último, las conclusiones.

2. METODOLOGÍA EDUCATIVA

En este apartado se presenta el enfoque docente dado al trabajo. La metodología aplicada en el desarrollo de esta actividad es el Aprendizaje Basado en Proyectos [6, 11], mediante la cual se explican de forma teórico-práctica los conceptos subyacentes a la técnica SLAM para, posteriormente, ejecutar el proyecto.

Partiendo de la premisa de que la mejor manera de entender el funcionamiento del SLAM es programando SLAM y conociendo los agentes que intervienen en esta técnica, el Aprendizaje Basado en Proyectos es una metodología adecuada por sus características para este tipo de actividades [10]:

- Se centra en el alumno y potencia su creatividad mediante la búsqueda de soluciones alternativas [1, 8].
- Favorece el análisis, la observación y la toma de decisiones.
- Construye el conocimiento y fomenta el aprendizaje a partir de la experiencia [12].
- Motiva al alumno, que participa de forma activa, y lo prepara para su futuro por su similitud con la realidad [2].
- Promueve el trabajo en equipo, habilidad demandada en el mercado laboral actual.

El proyecto se lleva a cabo durante la parte final de la asignatura. Aunque la planificación temporal puede sufrir modificaciones al tener que adaptarse a las necesidades de los alumnos, es importante definir una temporalización orientativa para que los estudiantes tengan una referencia y sepan los hitos que deben ir alcanzando.

Todo lo que involucra al proyecto se plantea para cinco semanas, empleando las dos sesiones semanales disponibles, cada una de dos horas. Es decir, en total, se invierten 20 horas lectivas. Fuera de este horario, el alumno puede trabajar por su cuenta las partes que considere oportunas, ya que los laboratorios y los materiales están a su disposición.

La propuesta consiste en dividir a la clase en grupos de tres personas, de tal forma que cada uno de los miembros se especializa en una parte del proyecto: 1) MATLAB, 2) ROS, 3) puesta en marcha del lidar y de la Raspberry. De forma paralela, el profesor explica las nociones básicas de SLAM para toda la clase.

Tras unas sesiones de estudio del tema asignado, el cual se completa con explicaciones y resolución de dudas por parte del profesor, deben ponerlo en común con el resto de compañeros de su grupo, para que el conocimiento sea compartido. Además, a través de esta dinámica, se alcanza una de las máximas dentro del ámbito educativo: “un alumno entiende de verdad un tema cuando es capaz de explicarlo con rigor”.

En la Figura 2 se muestra la temporalización aproximada planteada.

	Semana 1		Semana 2		Semana 3		Semana 4		Semana 5	
Fase Sesión	1	2	3	4	5	6	7	8	9	10
Nociones básica SLAM										
Toolbox MATLAB										
ROS										
Lidar y Raspberry										
Puesta en común										
Pruebas iniciales										
Pruebas finales										

Figura 2: Temporalización para el proyecto.

Las sesiones de programación de código y las pruebas iniciales se desarrollan en los laboratorios L-125 del Edificio Politécnico y L-102 Investigación Operativa de la Escuela Técnica Superior de Ingeniería Industrial de la Universidad de La Rioja.

3. PUESTA EN MARCHA DEL SISTEMA

En esta sección se presentan las herramientas, tanto *hardware* como *software*, empleadas a lo largo del proyecto.

3.1. HERRAMIENTAS EMPLEADAS

RPLIDAR A2

Se utiliza un sensor LIDAR, dispositivo que permite medir la distancia entre un emisor láser y un objeto o superficie utilizando un haz láser pulsado. Dicha distancia se determina midiendo el tiempo de retraso entre la emisión del pulso y su detección tras ser reflejado por el objeto.

En concreto el modelo empleado en este proyecto es el RPLIDAR A2 de Slamtec [14, 17] (Figura 3), adecuado para colocar en pequeños vehículos por su reducido tamaño y peso (190 g).

Su principal ventaja es el bajo coste en comparación con otros modelos disponibles en el mercado y, aunque no se trata de la opción más profesional, es una solución válida para pequeños proyectos e iniciarse en esta temática.

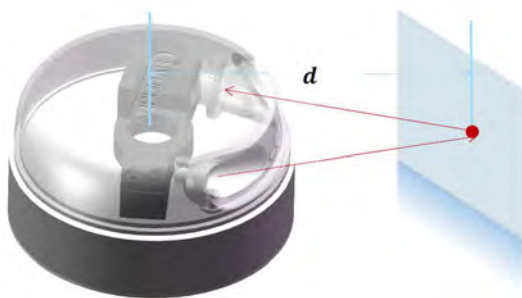


Figura 3: Funcionamiento del RPLIDAR A2 [17].

Raspberry Pi

Para las pruebas en campo se emplea una Raspberry Pi 3 Model B [15], con Ubuntu 16.04 versión Xenial instalado como sistema operativo, para poder trabajar con ROS. A este ordenador de placa reducida se conecta el LIDAR para recibir alimentación y adquirir los datos del entorno con el vehículo en movimiento.

Ambos dispositivos, gracias a su tamaño, se colocan sobre la plataforma móvil. En el que caso de este proyecto de iniciación se eligen dos coches teledirigidos (Figura 4), para facilitar la ejecución de las pruebas al no necesitar programación adicional, como sí ocurre con robots, centrándose, de este modo, en la propia implementación de SLAM.

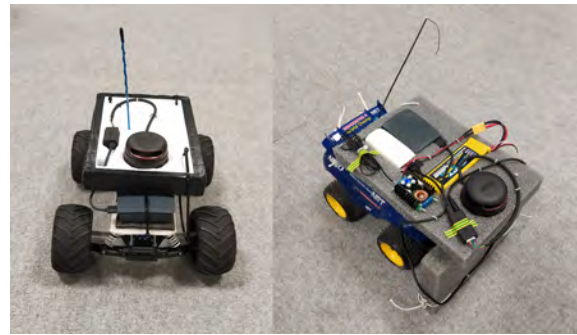


Figura 4: Plataformas móviles empleadas.

ROS

Robot Operating System [7, 13] es un *framework opensource* para el desarrollo de aplicaciones robóticas que involucren sensores y actuadores. Su filosofía se basa en la reutilización de los trabajos, impulsando la colaboración y favoreciendo un desarrollo sencillo gracias a su escalabilidad.

Entre sus elementos se encuentran:

- *Master*: administrador de ROS.
- Paquetes: unidades encargadas de organizar el *software*.
- Nodos: programas ejecutables asociados a procesos del robot.
- *Topics*: canales de comunicación utilizados por los nodos para intercambiar mensajes.
- Mensajes: estructura de datos para la comunicación entre nodos con un sistema de publicación y suscripción.

En la Figura 5 se presenta un diagrama de flujo explicativo sobre el funcionamiento de los nodos y los *topics*.

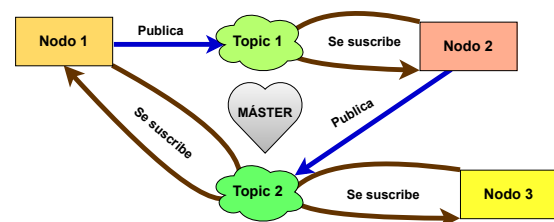


Figura 5: Diagrama del funcionamiento de ROS.

MATLAB

Es un *software* de programación, conocido por los alumnos de 4º de carrera en cualquier rama de ingeniería, que ofrece un entorno de desarrollo integrado con un lenguaje propio. Su código se implementa mediante funciones, clases y objetos y sus funcionalidades pueden ampliarse mediante librerías, denominadas *toolbox*.

En el proyecto se utiliza la versión R2019b, ya que es la primera que incorpora las dos *toolbox* requeridas [9], comentadas a continuación.

- *Navigation Toolbox* proporciona una serie de algoritmos y herramientas para implementar aplicaciones de movimiento y navegación. Entre las funcionalidades que interesan destacan: modelos de sensores, algoritmos de estimación de posición, representación de mapas en 2D y localización y mapeo simultáneo.
- ROS *Toolbox* facilita una interfaz que conecta MATLAB con ROS, lo cual permite desarrollar una red común de nodos, *topics* y mensajes. También dispone de funciones para importar y reproducir datos guardados en ROS en archivos *.bag*.

3.2. ELECTRÓNICA COMPLEMENTARIA

Tal y como se ha comentado previamente, la Raspberry Pi 3 se emplea como dispositivo al cual conectar el lidar para proporcionarle alimentación y obtener los datos del entorno con el vehículo en movimiento, por lo que es necesario que su sistema operativo sea Ubuntu para poder instalar ROS.

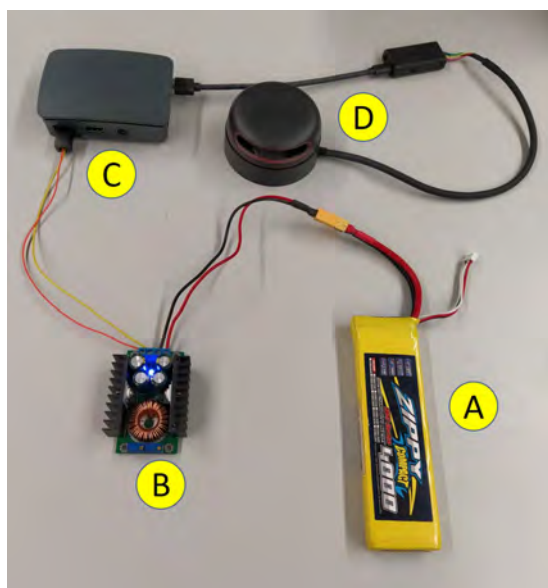


Figura 6: Batería LiPo (A), reductor DC/DC (B), Raspberry (C) y RPLIDAR A2 (D).

En este sentido, es importante tener en cuenta la alimentación del dispositivo, ya que al colocarse sobre una plataforma que se desplaza no es posible utilizar el puerto microUSB y un cargador tradicional. Para solventar este problema, se evalúan los requerimientos de voltaje y amperaje máximo de la Raspberry y del lidar.

Tras plantear un debate en el aula acerca de una solución técnica viable y simple, se opta por utilizar una batería LiPo de dos celdas y un reductor DC/DC como elemento de protección (Figura 6). Este reductor incorpora dos potenciómetros para regular los valores de salida según interese, lo cual asegura la alimentación adecuada para un correcto funcionamiento del sistema.

4. DESARROLLO TÉCNICO

En este apartado se presenta el código implementado en MATLAB para ejecutar SLAM.

4.1. ALGORITMO DE PROGRAMACIÓN EN MATLAB

En primer lugar, se explica de manera simplificada dicho código.

1. Definir la IP del equipo donde se ejecuta el *master* de ROS.
2. Establecer la conexión con ROS \Rightarrow comando *rosinit*.
3. Suscribirse al *topic* en el cual se publican los datos \Rightarrow instrucción *rossubscribe*.
4. Crear el objeto *lidarSLAM* y definir sus parámetros: resolución del mapa de rejilla de ocupación (celdas por metro), alcance del lidar (m), umbral de coincidencia de puntos y radio de búsqueda de puntos (m).
5. Definir un bucle *while* para la adquisición de los datos en tiempo real.
6. Extraer del subscriptor cada medida nueva publicada mediante la función *receive*.
7. Convertir los datos al tipo *lidarScan* e incorporarlos a *lidarSLAM* \Rightarrow comando *addScan*.
8. Mostrar el mapa de contorno y la trayectoria del vehículo.
9. Generar el mapa de rejilla de ocupación \Rightarrow funciones *scansAndPoses* y *buildMap*.
10. Desconectarse de ROS \Rightarrow comando *roshutdn*.

También es necesario saber que el algoritmo utilizado internamente por MATLAB para implementar SLAM es Pose Graph [4].

4.2. SLAM APP

Cuando los alumnos han entendido la programación que subyace la ejecución de SLAM, se propone el desarrollo de una aplicación mediante la herramienta App Designer, propia de MATLAB, para facilitar la ejecución de las pruebas y la parametrización de los valores más relevantes del algoritmo. Esta aplicación presenta una interfaz intuitiva con botones y elementos cuya programación interna ejecuta las mismas funciones que el código, pero desde un entorno visual y simplificado (Figura 7).



Figura 7: Interfaz de la aplicación.

A continuación, se indican algunos de estos elementos y la función que realizan:

- *Edit Field Text*: cuadro de texto para definir la IP del equipo donde se ejecuta ROS.
- *Toggle Switch*: definir estados asociados a la conexión y a la desconexión con ROS según la dirección IP anterior.
- *Drop Down*: seleccionar entre los *topics* disponibles.
- *Edit Field Number*: introducir parámetros del objeto de SLAM.
- *Slider*: seleccionar los otros dos parámetros necesarios en el objeto.
- *Button*: se utilizan tres → suscribirse al *topic*, crear el objeto *lidarSLAM* y detener el bucle sin finalizar la ejecución del programa.

5. RESULTADOS

En este apartado se presentan los mapas obtenidos en algunas de las pruebas. Las primeras pruebas se desarrollaron con obstáculos colocados en el L-102. Tras conseguir resultados satisfactorios, se proponen entornos más realistas, llevándose a cabo nuevas pruebas en diferentes espacios (Figura 8) de la Escuela Técnica Superior de Ingeniería Industrial de la Universidad de La Rioja.

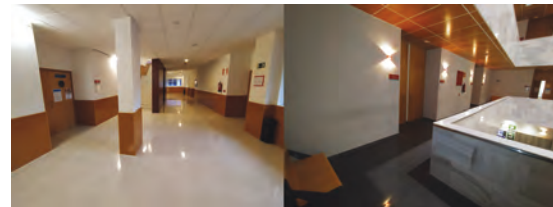


Figura 8: Espacios donde se desarrollan las pruebas: Zona 1 (izq.) y Zona 2 (dcha.).

Zona 1: Primera planta del Edificio Ampliación del Politécnico

En concreto se trata de la zona sureste, cerca del Laboratorio 102 - Investigación Operativa. La Figura 9 (arriba) muestra el mapa de contorno junto con el recorrido realizado, que consiste en dar una vuelta por el rellano y volver al punto de partida. En este mapa se muestran los escaneos realizados, los cuales definen los elementos que encuentra el vehículo a su paso: paredes, columna situada en (-15, 3), pilares, puertas y papeleras, como salientes o entrantes en la pared, ascensor en (-4, 5) y escaleras.

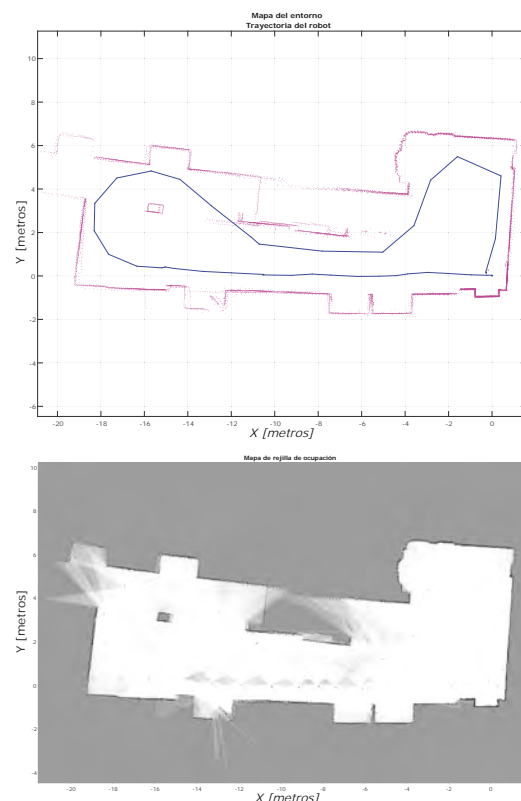


Figura 9: Resultados de la Zona 1.

En este último elemento se diferencian dos partes: por un lado, el tramo ascendente en $(-11, 4)$, del cual el lidar observa el escalón que está a su altura, y, por otro lado, el tramo descendente en las coordenadas $(-7, 3)$ que se representa como un hueco, ya que los escalones quedan por debajo del nivel del sensor.

El mapa de rejilla de ocupación (Figura 9 abajo) presenta un color blanco intenso por gran parte de su superficie ya que las distancias entre las paredes son menores al rango del lidar y la trayectoria ha cubierto todas las zonas, sin dejar puntos alejados o escondidos detrás de obstáculos.

Zona 2: Primera planta del Edificio Departamental

Esta segunda prueba se realiza en la parte más próxima al Laboratorio 111. La Figura 10 (arriba) presenta el mapa de contorno junto con la trayectoria del vehículo. En él se observan características que determinan buena calidad en el resultado obtenido, como los bancos del pasillo, en $(1, 0)$ y $(-9, -1)$, la puerta del laboratorio 111, coordenadas $(-11, 1)$, el radiador ubicado en $(-10, 3)$ o el pilar cercano a las escaleras, $(-8,5, 4,5)$.

Sin embargo, también se aprecian algunos errores: al comenzar la prueba, se tarda unos segundos en mover el vehículo y los datos almacenados se pintan de manera incorrecta, coordenadas $(0, -1)$ y $(2,5, 0)$, al no tener, todavía, ninguna referencia del entorno. Los huecos de puertas de despachos tampoco se muestran correctamente, ya que se dibuja una línea inexistente en la realidad.

El mapa de rejilla de ocupación (Figura 10 abajo) presenta un color blanco intenso por toda su superficie, ya que la anchura del pasillo es menor al rango del lidar.

6. CONCLUSIONES

En base a los resultados técnicos mostrados se puede afirmar que las expectativas iniciales de introducir a los alumnos en SLAM se ha cumplido, ya que se ha presentado un proyecto atractivo con resultados satisfactorios.

Como se ha podido comprobar durante el desarrollo del trabajo, la ejecución de SLAM en un sistema real es una tarea compleja, sin embargo, la utilización del Aprendizaje Basado en Proyectos y compartir el conocimiento entre compañeros con la estrategia de puzzle simplifica el proceso de enseñanza-aprendizaje. De esta forma se abarcan más conocimientos en menos tiempo y se adquieren los conceptos básicos para conocer una técnica tan relevante en la actualidad.

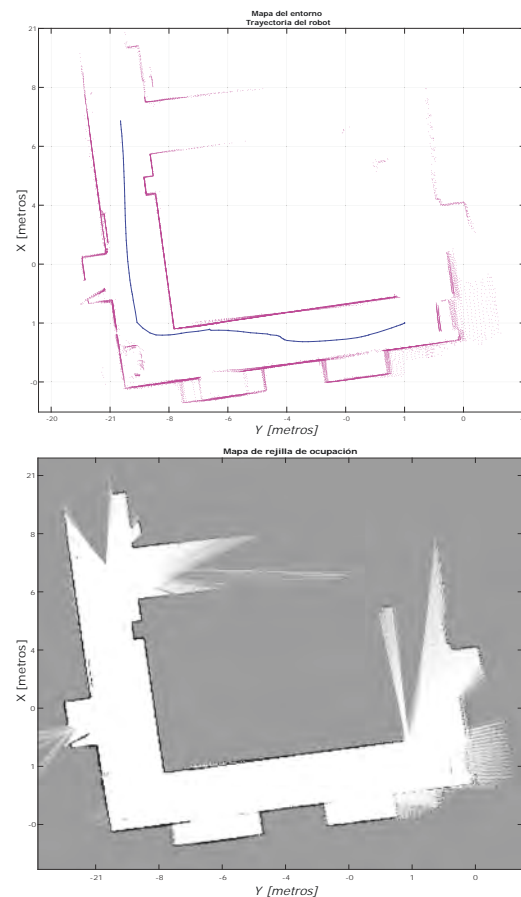


Figura 10: Resultados de la Zona 2.

Este trabajo ha permitido manejar plataformas *opensource*, como ROS y Raspberry, que, a menudo, quedan al margen del currículo académico, pero las cuales, gracias a sus características, facilitan el progreso técnico y aportan soluciones casi inmediatas a problemas experimentados anteriormente por otros usuarios de la comunidad. En cuanto a MATLAB se ha profundizado en el estudio de sus librerías.

También destacar la puesta en marcha de un sensor, el lidar RPLIDAR A2, cometido novedoso para el alumnado, pero a la orden del día en el ámbito industrial. Para conseguir este propósito, ha sido necesario analizar documentación técnica para conocer sus características y especificaciones.

Por estos motivos, se considera que la actividad ha sido un éxito y es previsible mantener su implantación en la asignatura mencionada a través del Aprendizaje Basado en Proyectos.

Agradecimientos

Los autores agradecen la ayuda prestada por la Universidad de La Rioja a través de las ayudas a la realización de tesis doctorales ATUR 2021 financiadas por Santander Universidades.

English summary

PROJECT-BASED LEARNING AS METHODOLOGY FOR AN APPROACH TO SLAM

Abstract

This work shows a first contact to university students with SLAM through the Project-Based Learning (PBL) methodology. SLAM is a technique whose objective is to provide a mobile robot with the ability to locate and position itself in an unknown environment, creating the map while it is navigating along it. The teaching approach that has been given to the field in the subject is also presented, as well as basic theoretical concepts from which they will develop a simple project. As for the project itself, the implemented application runs SLAM using the MATLAB Navigation Toolbox and ROS Toolbox together with ROS, a specific work environment for robots, which is installed on a Raspberry Pi 3. With this device of autonomy we can provide autonomy to two land mobile platforms that incorporate a lidar-type laser sensor to obtain data from the environment.

Keywords: PBL, SLAM, Mobile robot, MATLAB, ROS, Lidar.

Referencias

- [1] Barak, M., Yuan, S., 2021. A cultural perspective to project-based learning and the cultivation of innovative thinking, *Thinking Skills and Creativity*.
- [2] Chidhachack, S., Schulte, M. A., Ntow, F. D., Lin, J.-L., Moore, T. J., 2013. Engineering Students Learn ABET Professional Skills: A Comparative Study of Project-Based-Learning (PBL) versus Traditional Students, *The 2013 ASEE North Midwest Section Conference*, 147–166.
- [3] Corke, P., 2017. *Robotics, Vision and Control: Fundamental Algorithms in MATLAB*, Springer.

- [4] Deutsch, I., Liu, M., Siegwart, R., 2016. A framework for multi-robot pose graph SLAM, *IEEE International Conference on Real-Time Computing and Robotics*, 567–572.
- [5] Fernandez-Madriral, J.A., Blanco, J.L., 2013. *Simultaneous Localization and Mapping for Mobile Robots: Introduction and Methods*, Information Science Reference.
- [6] Kokotsaki, D., Menzies, V., Wiggins, A., 2016. Project-based learning: a review of the literature, *Improving schools*, 19(3), 267-277.
- [7] Koubaa, A., 2016. *Robot Operating System (ROS)*, Springer.
- [8] Lin, K. Y., Wu, Y. T., Hsu, Y. T., Williams, P. J., 2021. Effects of infusing the engineering design process into STEM project-based learning to develop preservice technology teachers' engineering design thinking, *International Journal of STEM Education*. 8(1), 1–15.
- [9] MathWorks, 2019. *Navigation Toolbox*. <https://es.mathworks.com/help/matlab>.
- [10] Mills, J. E., Treagust, D. F., 2003. Engineering education-Is problem-based or project-based learning the answer?, *Australasian Journal of Engineering Education*.
- [11] Ministerio de Educación Cultura y Deporte, 2015. *Aprendizaje Basado en Proyectos. Infantil, Primaria y Secundaria, Recursos Educativos*.
- [12] Moraes, P., Ferreira, M. J., Veloso, B., 2021. Improving Student Engagement with Project-Based Learning: A Case Study in Software Engineering, *Revista Iberoamericana de Tecnologías Del Aprendizaje*, 16(1), 21–28.
- [13] Open Source Robotics Foundation. *ROS*. <https://www.ros.org>.
- [14] Open Source Robotics Foundation. *rplidar*. <http://wiki.ros.org/rplidar>.
- [15] Raspberry Pi Foundation. *Raspberry Pi*. <https://www.raspberrypi.org/>.
- [16] Riisgaard, S., Blas, M.R., 2004. *SLAM for the dummies: A Tutorial Approach to Simultaneous Localization and Mapping*.
- [17] Shanghai Slamtec. *RPLIDAR A2*. <https://www.slamtec.com/en/Lidar/A2>.

- [18] Siegwart, R., Nourbakhs, I., 2004. Introduction to Autonomous Mobile Robot, The MIT Press.
- [19] Smith, R.C., Cheeseman, P., 1986. On the Representation and Estimation of Spatial Uncertainty, The International Journal of Robotics Research, 5(4), 56-68.
- [20] Thrun, S., Bugard, W., Fox, D., 2005. Probabilistic Robotics, The MIT Press.



© 2022 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>).