

UNIVERSIDAD DE BUENOS AIRES
FACULTAD DE INGENIERÍA
CARRERA DE ESPECIALIZACIÓN EN SISTEMAS
EMBEBIDOS



MEMORIA DEL TRABAJO FINAL

**Firmware para robot autónomo
utilizando FreeRTOS**

Autor:
Alexis Martin Pojomovsky

Director:
Dr. Pablo De Cristóforis (FCEyN-UBA)

Jurados:
Esp. Ing. Diego Fernández (FI-UBA)
Esp. Ing. Edgardo Comas (CITEDEF/UTN-FRBA)
Esp. Ing. Gerardo Puga (UNLP)

*Este trabajo fue realizado en las Ciudad Autónoma de Buenos Aires, entre enero
de 2018 y diciembre de 2019.*

Resumen

En el presente trabajo se describe el desarrollo de un robot móvil destinado a utilizarse por estudiantes universitarios en el aprendizaje de tópicos de robótica, tales como odometría, localización y navegación.

Esta propuesta se ajusta a dos importantes restricciones: disponibilidad de componentes en el mercado local y bajo precio, siendo esta última relativa a productos comerciales de similares características.

Se incluye además, un repositorio público en Github desde el que se puede acceder al software de ejemplo y documentación necesaria para comenzar a utilizar la plataforma.

Agradecimientos

Redactar
agradeci-
mientos.

Índice general

Resumen	III
1. Introducción general	1
1.1. Robótica móvil	1
1.2. Motivación	2
1.3. Descripción de tecnologías	3
1.4. Objetivos y alcance	3
2. Conclusiones	5
2.1. Conclusiones generales	5
2.2. Trabajo a futuro	6

Índice de figuras

1.1. Algunas de las posibles trayectorias que podría adoptar el robot móvil.	2
---	---

Índice de Tablas

Dedicado a... [OPCIONAL]

Capítulo 1

Introducción general

En este capítulo se introduce el campo de estudio de la robótica, contraste entre robótica de manipuladores y móvil, así como la importancia de una planta de pruebas física como motivación para la realización de este trabajo. Asimismo, se presentan los objetivos y el alcance del presente proyecto.

1.1. Robótica móvil

La robótica de manipuladores, también llamados brazos robóticos, se han ganado su puesto como ciudadanos de primera clase en la industria de la manufactura. Dificilmente podríamos al día de hoy, imaginarnos una planta de fabricación en serie que no disponga de estos dispositivos para la realización de tareas repetitivas y de alta precisión.

En la industria electrónica, por citar un ejemplo, los manipuladores son capaces de colocar componentes de montaje superficial con una precisión y velocidad por lejos sobre-humana, haciendo posible la elaboración de teléfonos celulares, computadoras portátiles, etc. Sin embargo, y a pesar de su innegable éxito, estos robots sufren de una desventaja particular: la falta de movilidad. En contraste con un manipulador fijo posee un rango de movimiento limitado que depende del sitio en que el mismo se encuentre instalado, un robot móvil es capaz de moverse a través de la planta, permitiendo el aprovechamiento de sus facultades donde sea que estas sean precisadas. Ante estas limitaciones, en los años noventa surgen los robots móviles.

Una definición correcta de robot móvil plantea la capacidad de movimiento sobre entornos no estructurados, de los que se posee un conocimiento incierto, mediante la interpretación de la información suministrada a través de sus sensores y del estado actual del vehículo.

El principal problema a resolver en un robot móvil es el de generar trayectorias y guiar su movimiento según éstas, en base a la información proveniente de los sensores externos, permitiendo al vehículo desplazarse entre dos puntos cualesquiera del ambiente de trabajo de manera segura, sin colisiones. Esto exige diseñar sistemas de control de trayectorias (posición, dirección, velocidad) en diversos niveles jerárquicos, de manera que el procesamiento de la información proveniente de los sensores externos asegure la mayor autonomía posible.

Los robots móviles operando en grandes ambientes no estructurados deben enfrentarse a significativas incertidumbres en la posición e identificación de objetos. En efecto, la incertidumbre es tal que, trasladarse desde un punto A hasta un punto B es una actividad arriesgada para un robot móvil, una actividad relativamente trivial para un manipulador industrial. En compensación por tener que enfrentarse con más incertidumbres del entorno, no se espera que un robot móvil siga trayectorias o alcance su destino final con el mismo nivel de precisión que se espera de un manipulador industrial (en el orden de las centésimas de milímetro).

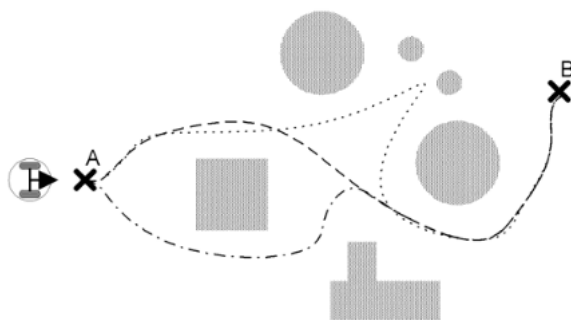


FIGURA 1.1: Algunas de las posibles trayectorias que podría adoptar el robot móvil.

1.2. Motivación

El estudio de la robótica en las universidades de la región se encuentra mayormente avocada a la robótica de manipuladores. Esto tiene sentido desde un punto de vista de oferta y demanda en la industria local, sin embargo, el mercado internacional esta viviendo una fuerte demanda de profesionales capaces de entender y aplicar técnicas de robótica móvil.

Teniendo en cuenta que hasta hace solo un puñado de años atrás, los dispositivos requeridos para llevar a cabo estas tareas hacían que el estudio, desarrollo y aplicación de tareas de robótica móvil sean extremadamente costosos, aún a pequeña escala, la realidad actual es diferente. Hoy, y gracias a la masificación de ciertas tecnologías, es posible armarse de toda una gama de sensores, actuadores y sistemas capaces de procesar el flujo de datos requerido para estas aplicaciones con muy poco dinero, en muchos casos reutilizando componentes originalmente destinados a otros propósitos.

Aún con la importante reducción de precios vivida en los últimos años, las plantas de prueba ofrecidas por empresas internacionales para fines didácticos continúan siendo prohibitivas para muchas instituciones educativas, así como también para la mayoría de los estudiantes. Es por esto que el presente trabajo se desarrolla en torno a la propuesta de planta de pruebas para el estudio y aplicación de técnicas de robótica móvil. Esta propuesta se centra en dos restricciones importantes, bajo costo y disponibilidad en el mercado local.

1.3. Descripción de tecnologías

Blabla.

1.4. Objetivos y alcance

Blabla.

Capítulo 2

Conclusiones

En este capítulo se describen los aportes generados con el trabajo realizado, detallando los logros obtenidos. Así también, se especifican las técnicas mediante las cuales se hizo posible la ejecución del proyecto. Por último, se deja constancia de las metas originales que no pudieron ser abordadas dentro del alcance final, identificando las propuestas de acción a futuro.

2.1. Conclusiones generales

En este trabajo se completó la primer iteración en el ciclo de diseño e implementación de una planta de pruebas destinada a utilizarse en el aprendizaje de técnicas de robótica móvil a nivel personal o universitario.

Se pudo desarrollar una planta de pruebas completa y funcional en base a componentes disponibles en el mercado local, integrándola con el framework de robótica ROS. Se pudo desarrollar un firmware capaz de funcionar como una interfaz entre una base móvil comercial y ROS, así como interfacear los sensores necesarios con dicho framework de modo a posibilitar su uso inmediato en aplicaciones de navegación autónoma. Este desarrollo sienta las bases y estructura para una segunda iteración, dejando abiertas una serie de propuestas con mejoras que pueden implementarse en base a las necesidades que se le vayan presentando al usuario.

A lo largo del desarrollo de este trabajo final, se aplicaron extensivamente los conocimientos adquiridos durante la carrera. Mas allá de que el conjunto de asignaturas posibilitaron la apreciación de un panorama muy completo en lo que refiere a diseño y ejecución de un proyecto de sistemas embebidos, se destacan a continuación aquellas materias cuyo impacto en el desarrollo de este trabajo:

- **Protocolos de Comunicación:** se aplicaron los conocimientos adquiridos en la utilización de protocolos UART e I2C. Así también, se implementó la librería `roscpp`, la cual permite ser utilizada de manera intercambiable tanto con comunicación UART como Ethernet, utilizadas durante la etapa inicial y final del proyecto, respectivamente.
- **Sistemas Operativos de Tiempo Real (I y II):** se utilizaron los conocimientos adquiridos aplicando FreeRTOS como sistema operativo para el sistema propuesto.

- Arquitectura de microprocesadores: resultó necesario tener conocimientos sobre la Arquitectura ARM Cortex M para la programación de la plataforma STM32 NUCLEO y el uso efectivo de sus periféricos.
- Programación de microprocesadores: se aplicaron las buenas prácticas de programación que fueron mostradas a lo largo de la materia. Se empleó un formato de código consistente a modo de obtener código más modular, legible y reutilizable.
- Gestión de Proyectos en Ingeniería: la elaboración de un Plan de Proyecto para organizar el Trabajo Final, el cual facilitó en gran manera la ejecución del mismo y evitó demoras innecesarias.

Por lo tanto, se concluye que los objetivos planteados al comienzo del trabajo han sido alcanzados satisfactoriamente, habiéndose cumplido con los criterios de aceptación del sistema final y obtenido conocimientos valiosos para la formación profesional del autor.

2.2. Trabajo a futuro

Se plantea a continuación una lista de mejoras que podrían implementarse a fin de aumentar las capacidades del robot, las cuales se espera puedan ser abordadas por los usuarios que hayan decidido adoptar la plataforma, a quienes se exhorta a hacer aportes públicos al repositorio de modo a que todos puedan verse beneficiados con los mismos:

- Agregar soporte para sensores y actuadores disponibles en el robot Roomba que aún no han sido implementados en el sistema.
- Agregar un magnetómetro a la IMU via SPI o bien, reemplazarla por una IMU que ofrezca 9 grados de libertad, como la MPU9250, de modo a que la estimación de la pose del robot sea mas precisa.
- Utilizar la salida de potencia del controlador del motor de la aspiradora para alimentar el sensor Kinect, eliminando la necesidad de una batería externa.
- Reemplazar el actual computador OrangePI PC con un Nvidia Jetson Nano o similar, de modo a poder procesar la información proveniente desde una mayor cantidad de sensores.
- Implementar un segundo sensor Kinect apuntando en dirección contraria al actual para duplicar el campo de visión.
- Agregar soporte para ROS 2.