

# Capítulo 5

---

Conclusiones y Desarrollos Futuros

---

## 5. CONCLUSIONES Y DESARROLLOS FUTUROS

---

En este capítulo se detallan las conclusiones a las que se han llegado tras la realización de este Proyecto Fin de Carrera, así como un conjunto de actuaciones que se proponen como futuras líneas de actuación.

### 5.1. Conclusiones

---

Este Proyecto Fin de Carrera ha permitido incluir en el ROMEO-4R la nueva arquitectura software que se define en el documento “*Arquitectura para múltiples robots heterogéneos*” y que actualmente se está implantando en varios robots móviles. Esta arquitectura pretende, entre otras cosas, mejorar la cooperación entre aquellos robots que se encuentren bajo la misma, así como facilitar la adición de nuevos robots al sistema.

La creación de la *Robot Implementation Layer* ha permitido por tanto completar los cimientos de esta nueva arquitectura, dando lugar al perfecto funcionamiento del vehículo en la realidad. Por otra parte, la implementación de esta capa, aunque dependa del robot en la que se use, facilitará la creación de la misma en otros robots distintos al ROMEO-4R, ya que las pautas de programación serán las mismas.

Este proyecto comenzó con la implementación del *Hardware Abstraction Module*, añadiendo uno a uno todos los dispositivos con los que cuenta el vehículo, para lo cual se aprovechó parte del software recopilado de varios Proyectos Fin de Carrera. Todo este código fue depurado y adaptado a la nueva arquitectura, lo que requirió un gran esfuerzo debido a la necesidad de conocer en profundidad la mayoría de las funciones implementadas por tantas personas.

Tras realizar las pruebas pertinentes y comprobar que todos los dispositivos del ROMEO-4R funcionaban a la perfección, se procedió a realizar la siguiente fase del Proyecto Fin de Carrera que consistía en la realización del *Romeo Status Module* y del *Path Follower Module*, que permitían cerrar el bucle de control en un principio. Para el desarrollo de éste último se implementó un seguimiento de caminos basado en el método de persecución pura.

Posteriormente, para poder realizar las pruebas de software oportunas, se creó un simulador del hardware que hacía las funciones del *Hardware Abstraction Module* de una forma muy próxima a la realidad, para lo cual se implementó un modelo cinemático y dinámico bastante acertado. Con este simulador, se obtuvieron grandes resultados que hubiesen sido imposibles de conseguir trabajando directamente con el vehículo real, además del peligro para la integridad física del mismo.

Después de conseguir el perfecto funcionamiento en simulación del vehículo, introduciéndole nosotros la trayectoria mediante un programa de pruebas, se procedió al desarrollo del *Trajectory Generation Module*, lo cual nos permitió completar la arquitectura mediante la posibilidad de asignar al vehículo trayectorias más complejas.

La fase final del proyecto fue la puesta a punto del ROMEO-4R en la realidad, sustituyendo el módulo de simulación del hardware por el módulo que trabaja con el hardware en sí. Los resultados obtenidos fueron excelentes tanto a nivel del propio ROMEO-4R, como a nivel de cooperación con múltiples robots heterogéneos, como se recoge en el Proyecto presentado por Antidio Viguria Jiménez, así como en el resto de pruebas posteriores.

Para concluir, se deja constancia de que, por primera vez, se ha conseguido ver en un robot el funcionamiento real de esta nueva arquitectura, con la satisfacción correspondiente al ver que el trabajo de todo el *Grupo de Robótica Visión y Control* había dado los frutos deseados.

Ahora, el trabajo debe continuar para mejorar dicha arquitectura, para lo cual en el siguiente apartado se exponen los aspectos más destacados sobre los que trabajar en el futuro, y para desarrollar nuevas líneas de investigación en el campo de múltiples robots.

## 5.2. Desarrollos Futuros

---

Aunque el ROMEO-4R se encuentra ya en funcionamiento con la nueva arquitectura software, y por tanto la parte más compleja de la misma ya está realizada, existen una gran variedad de actuaciones a desarrollar, entre las que destacan:

- **Sistema de visión:** desarrollar los módulos de percepción para poder integrarlos en esta arquitectura, permitiendo que la información que nos ofrezca la misma pueda ser utilizada por el resto de módulos y capas superiores.
- **Pan&Tilt:** desarrollar el módulo que se encargue de controlar el *pan&tilt* para las cámaras de visión.
- **Algoritmos:** implementar los algoritmos necesarios que permitan el desarrollo de nuevas funcionalidades en el uso de los dispositivos existentes, lo cual permitirá la realización de tareas más complejas.
- **Seguimiento de caminos:** implementar la gran variedad de métodos que existen en lo que respecta al seguimiento de caminos, para lo cual, básicamente, lo único necesario será cambiar la clase existente por la nueva.
- **Generador de trayectorias:** implementar los distintos métodos de generación de trayectorias, sustituyendo el método actual (trayectorias en líneas rectas), por métodos más completos, como la utilización de curvas  *$\beta$ -spline*. Al igual que en el seguimiento de caminos, esto se consigue fácilmente con sólo desarrollar la nueva clase y sustituirla por la existente, sin necesidad de realizar modificaciones de mayor importancia.

- **Simulador:** desarrollar en el simulador los archivos correspondientes a los sonares y al láser, para que se tenga un enfoque más realista de los mismos.
- **Estado del ROMEO-4R:** desarrollar la política de gestión para la elección de los parámetros que se usarán para rellenar los campos del estado del ROMEO-4R en función de los dispositivos (sensores) disponibles. Esto debe incluirse en la función `update_romeo_state()` que se encuentra en `./romeo/ril/romeo_status_module/source/romeoRSM.cpp`.
- **Tratamiento de errores:** desarrollar todo lo referente a la gestión de errores que se produzcan, prestando una especial atención a aquellos que dan lugar al fallo completo del sistema para que en caso de ocurrir, el vehículo finalice de forma segura, evitando cualquier percance físico.
- **Investigación:** con el desarrollo de esta arquitectura se ha conseguido una excelente plataforma de investigación en el campo de la cooperación de robots, por lo que se puede trabajar en el desarrollo de software que posibilite la interacción entre múltiples robots.