

Arquitectura dual-modular para desarrollos y validación de módulos de decisión y control en vehículos automatizados

R. Lattarulo^{a, *}, J. A. Matute^a, J. Pérez^a, V. Gomez Garay^b

^aTecnalia Research and Innovation, Parque Científico y Tecnológico de Vizcaya, Calle Geldo, Edif 700, 48160, Derio, España.

^bUniversidad del País Vasco, Dpto. de Ingeniería de Sistemas y Automática. Escuela de Ingeniería de Bilbao, Plaza Ingeniero Torres Quevedo 1, 48013, Bilbao, España.

Resumen

El avance logrado durante las últimas décadas en los sistemas avanzados de asistencia a la conducción (ADAS, *Advanced Driver Assistance System*) ha posibilitado mejorar múltiples aspectos en los vehículos comerciales, como por ejemplo la seguridad, robustez de los sistemas, eficiencia energética, detección de peatones, aparcamiento asistido y ayudas a la navegación, entre otros. Algunos desarrollos, como el control lateral y la generación óptima de trayectorias en tiempo real, están en pleno desarrollo. En este trabajo se presenta una arquitectura dual-modular cuyas principales características son su capacidad para integrar y probar nuevos algoritmos de control y decisión (modular), y la posibilidad de llevar a cabo pruebas en entornos simulados y en plataformas reales (dual), reduciendo los tiempos y costes de desarrollo. Con esta arquitectura se han podido probar diferentes técnicas de control y de generación de trayectorias, realizando además simulaciones, y comparando los resultados obtenidos con un vehículo real.

Palabras Clave:

Automoción, Vehículos Automatizados e Inteligentes, Guiado, Arquitecturas de Control, Planificación y Seguimiento de Trayectorias, Modelado Multicuerpo.

Dual-modular architecture for developing and validation of decision and control modules for automated vehicles

Abstract

In last decades, the advances done in the Advanced Driver Assistance System (ADAS) have improved multiple aspects in the vehicles, as: safety, system robustness, power efficiency, pedestrian detection and road lanes, assisted parking, navigation, etc. In the other hand, lateral control and generation of optimal trajectories in real time, are under development. In this work, we present a dual modular architecture. Its principal characteristics are the capacity of integrate and test new control algorithms, and the possibility of making tests with the simulation environment and the real platform (dual), reducing the development time. This architecture has been used to test different techniques for control and trajectory generation. Furthermore, the simulations have been done with a high level of precision comparing them with a real vehicle.

Keywords:

Automotive, Intelligent and automated vehicles, Guidance, Control architecture, Planning and trajectory tracking, Multibody systems.

1. Introducción

La seguridad vial es un tema de actualidad que interesa por igual a conductores, peatones y administraciones públicas. Un estudio reciente ha evaluado que al menos 1.2 millones de per-

sonas mueren al año como producto de una colisión, estableciendo que en la gran mayoría de estos accidentes la causa determinante ha sido un error humano (The International Traffic Safety Data and Analysis Group IRTAD, 2017). El mismo grupo apunta a que el incremento del Producto Interior Bruto

* Autor para correspondencia: rayalejandro.lattarulo@tecnalia.com

(PIB) de muchos países ha promovido el aumento sustancial del número de vehículos en circulación y, por tanto, a elevar la previsión de siniestros. En otro estudio se detalla que una forma de reducir la magnitud de este problema es invertir en los sectores productivos adecuados (Bertoncello and Wee, 2015). Un posible sector es la conducción automatizada, estimando en un 90 % el número de accidentes que podrían ser evitados si se mejoran las investigaciones en este ámbito.

La evolución de la conducción automatizada se ha acelerado gracias a la base de conocimiento aportada por los Sistemas Avanzados de Asistencia a la Conducción (ADAS, *Advanced Driver Assistance System*). Muchos de estos sistemas se encuentran instalados en la gran mayoría de vehículos comerciales hoy en día. Un ejemplo de estos es el Sistema Anti-Bloqueo de Frenado (ABS, *Antilock Braking System*) en todos los vehículos nuevos. El ABS fue diseñado por Bosch en 1978 y es obligatorio en Europa desde 2004. Otro ejemplo es el Control de Velocidad Crucero (CC, *Cruise Control*), que desde hace varios años se instala en vehículos de gama media y alta. El CC constituye el primer paso en la automatización de la conducción a nivel longitudinal (Bengler et al., 2014; Milanese et al., 2012b). Todos los éxitos logrados en el sector han sido posibles gracias a la gran cantidad de proyectos, instituciones, consorcios y fundaciones (de carácter público y privado) que han alentado el desarrollo de estas iniciativas.

Entre las más relevantes se encuentran las primeras pruebas que hizo la Universidad Carnegie Mellon (CMU, *Carnegie Mellon University*), con un monovolumen Pontiac, en el año 1995 (Navlab 5). CMU automatizó el volante usando un sistema de visión (Jochem et al., 1995). Otro evento importante tuvo lugar en 1997 en el transcurso del Free Agent Demonstration (FDA), en la autopista I-15 de San Diego, California. En el FDA fue verificado el escenario de platooning y seguimiento de ruta haciendo uso de un sistema de visión artificial. Años más tarde, los DARPA Challenge de 2004, 2005, y 2007, representaron un salto cualitativo. En todos ellos se realizaron un gran número de pruebas: desde recorridos automáticos sobre vías marcadas y no marcadas, hasta llegadas a un punto de misión, seguimiento de trayectoria, detección de obstáculos, o el cumplimiento de las normas de tráfico (Ozguner et al., 2004; Thrun, 2006; Yoon and Crane, 2008; Kress-Gazit and Pappas, 2008; Woo et al., 2008).

En años más recientes, empresas como Google, con una flota de más de 80 vehículos, han llevado a cabo pruebas reportando únicamente un accidente cada 85000 kilómetros (Favaro et al., 2017). Por otra parte, Nvidia ha creado un ordenador (conocido como Drive PX2) especialmente diseñado para aplicaciones de conducción automatizada. Tesla, Mercedes Benz, Ford y General Motors (GM) se han sumado también a iniciativas en este campo. En el caso de Tesla, la empresa ha llegado a comercializar con éxito sus aplicaciones.

Si bien es cierto que se lleva trabajando varios años en esta temática, y que el grado de automatización de los vehículos se ha elevado considerablemente, existen muchos campos en los que es necesario profundizar. Tal es el caso de la generación dinámica de trayectorias y el control lateral de los vehículos.

El objetivo de este trabajo es presentar una arquitectura de control dual y modular. Dual (virtual/real) ya que se utiliza en pruebas con simulador y en un vehículo real sin generar cambios, y modular ya que se encuentra separada en unidades según

la tarea que desempeña. El enfoque principal es servir de base para desarrollos futuros en términos de: 1) validación de la generación dinámica de trayectorias y 2) comparativa entre técnicas de control lateral y longitudinal. La arquitectura se valida usando un ejemplo comparativo entre un vehículo virtual y uno real.

El resto del trabajo contiene un breve resumen del estado de arte en la sección 2. En la sección 3 se expone la arquitectura de control dual-modular, que consta de 7 bloques principales: adquisición de información, percepción, comunicaciones, módulo de generación de trayectorias, control, actuación e Interfaz Humano Máquina (IHM). En la sección 4 se explica el formato y herramientas utilizadas para construir la arquitectura. Seguidamente, en la sección 5, se presenta un ejemplo de pruebas realizadas considerando una comparativa de controladores laterales y la generación de trayectorias. Para realizar estas pruebas fueron utilizados los entornos virtual y real. Finalmente se abordan las conclusiones y la continuidad de los trabajos futuros en la sección 6.

2. Estado del arte

La mayoría de los trabajos que se han realizado en conducción automatizada utilizan esquemas de control específicos según la aplicación estudiada. De ahí que resulte necesario modificar constantemente la arquitectura con cada aplicación o escenario. En esta sección se presenta un breve resumen del estado del arte en arquitecturas de control para vehículos automatizados.

La arquitectura Robust Sense (Tas et al., 2018) se encuentra separada en subsistemas con los objetivos de aumentar la fiabilidad y robustez de los vehículos automatizados. Tiene como principal novedad realizar evaluaciones del funcionamiento general del sistema y de la percepción del entorno. La arquitectura tiene la capacidad de trabajar en operación degradada en caso de falla en algún subsistema o sensor. El principal problema es que ofrece un nivel de detalle bajo para los módulos de comunicación, decisión o control en comparación con el nivel de detalle usado en la percepción.

Debido al elevado esfuerzo necesario para realizar pruebas en vehículos reales se presentó una arquitectura escalable para vehículos automatizados (Berntorp et al., 2018). Los autores hicieron pruebas preliminares en un vehículo a escala como paso previo a su realización en un vehículo real. Esta arquitectura obtiene la información directamente de los sensores y únicamente se centra en las etapas de decisión, control y actuación. Al igual que el caso previo, no se considera la comunicación del vehículo con otros participantes. Por otra parte, tampoco se tiene en cuenta el bloque de modelado del vehículo y el entorno (percepción) en la arquitectura de alto nivel.

Una patente de arquitectura de control ha sido presentada recientemente (Klaus et al., 2018). El trabajo se enfoca en la maniobra de seguimiento de vehículos automatizados o platooning basado en comunicaciones. Cuenta principalmente con tres módulos: 1) el controlador que se encarga de mantener los vehículos ejecutando la maniobra de platooning, 2) el módulo de comunicaciones a través del que se recibe y envía la información necesaria durante la maniobra, y 3) la IHM junto con un módulo para monitorizar el estado del conductor.

Otra arquitectura modular para vehículos automatizados ha presentado las unidades de decisión y control como un único módulo (Jones et al., 2018). Los autores la han separado en tres niveles: 1) nivel estratégico que se encarga de decidir el camino a seguir de un punto de inicio a uno de finalización, 2) nivel táctico que se encarga de generar consignas para mantenerse o cambiar de carril, y 3) generación de la trayectoria y su seguimiento. Esta propuesta abarca de forma más completa la información faltante en los casos anteriores, pero la unión de los módulos de decisión y control no permite realizar pruebas de nuevos algoritmos con facilidad.

Como se puede apreciar ninguna de las arquitecturas expuestas anteriormente abarca todas las funcionalidades requeridas para el desarrollo y pruebas de algoritmos para vehículos automatizados en distintos escenarios.

3. Arquitectura dual-modular

La arquitectura de control para conducción automatizada propuesta en este trabajo permite la simulación de nuevos algoritmos de control y la realización de pruebas en distintos escenarios. Al mismo tiempo, permite una implementación en vehículos reales sin hacer modificaciones en los algoritmos probados.

La arquitectura es modular, facilitando la integración de nuevos algoritmos y funciones para la resolución de problemas que se plantean en conducción automatizada. La Figura 1 muestra los 7 bloques que la constituyen (González et al., 2016). El vehículo real y la simulación aparecen dibujados con líneas punteadas ya que son los receptores de la arquitectura.

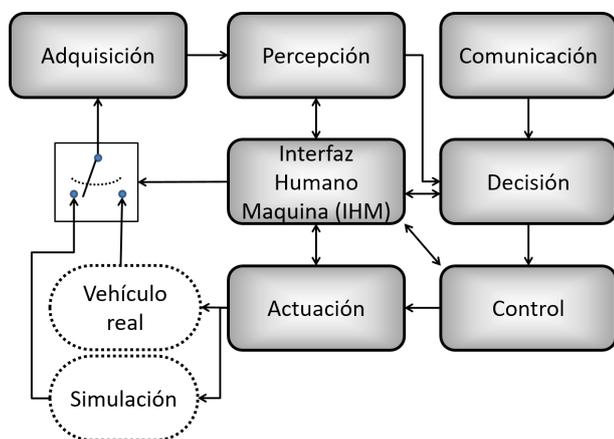


Figura 1: Planteamiento de la arquitectura de control

El subsistema de adquisición de información está constituido por cada uno de los sensores embarcados en el vehículo tanto en la plataforma real, como en la simulada y se encarga de leer los datos procedentes del vehículo, sea éste simulado o real, haciendo uso del IHM. Este módulo procesa la información y la transforma en unidades estandarizadas. Las variables de posición son las coordenadas UTM (*Universal Transverse Mercator*), y de forma adicional las de latitud y longitud, para registro. Las velocidades son leídas en los 3 ejes (longitudinal, lateral y vertical del vehículo) en este mismo módulo. La aceleración longitudinal y lateral junto con los ángulos de pitch, yaw y roll también son recolectadas al igual que la la odometría

(lectura de la posición del volante, velocidad del vehículo, entre otros).

El bloque de percepción combina la información entregada por el subsistema de adquisición a través de distintos sensores, como GPS, láser, LiDAR, radar, y ultrasonido permitiendo el posicionamiento en los mapas, la replanificación de rutas, la evasión de obstáculos y la generación de maniobras de adelantamiento. Esto conlleva a un aumento de la seguridad en el vehículo y del entorno. La unidad también realiza la fusión de datos de los distintos sensores con objeto de mejorar la calidad de las variables leídas en adquisición.

La unidad de comunicaciones ofrece información detallada del medio que rodea al vehículo. Cuenta con la capacidad de comunicarse con otros vehículos o V2V (Harding et al., 2014), con personas o comunicación V2P (Bagheri et al., 2014) y señales inteligentes o unidades al lado de la carretera (RSU, *Road Side Unit*) conocida también como V2I (Milanes et al., 2012a).

El bloque de decisión planifica las trayectorias que seguirá el vehículo y se compone de 3 subsistemas. Esta división garantiza la modularidad de las tareas realizadas en términos de las trayectorias generadas, como son: 1) creación del mapa base, 2) la generación de trayectorias continuas, 3) capacidad de recalcular trayectorias en caso de eventos inesperados como adelantamientos, cambios de carril, paradas de emergencia y cambio de destino (Figura 2).

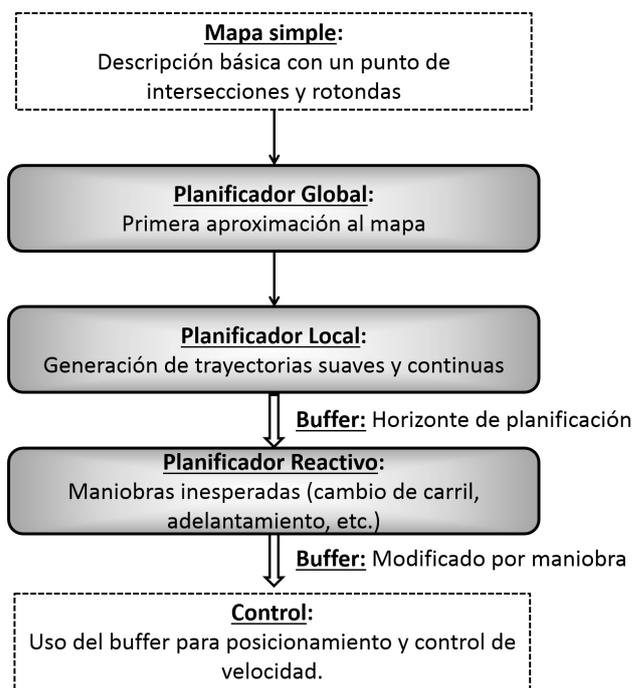


Figura 2: Los 3 subsistemas involucrados en decisión.

El planificador global toma la información procedente de un mapa sencillo descrito con puntos (en la Figura 3 se muestra un ejemplo de este tipo de mapa) y genera como respuesta una primera aproximación a un mapa global (ver Figura 4). Los tipos básicos de puntos utilizados son: tipo 1 que se refiere a una intersección. Las coordenadas X y Y son su ubicación y las demás características no son utilizadas. El tipo 2 se usa para caracterizar las rotondas, donde las coordenadas X e Y

indican la posición de su centro, la curvatura describe el radio de la rotonda y los valores de ángulo de entrada y ángulo de salida, describen las rectas que interceptan a la rotonda.

```
// Intersección
Nodo 1 </
X      : 0.000 (Posición en eje X)
Y      : 0.000 (Posición en eje Y)
Max. Speed: 30.000 (Velocidad Máxima del tramo)
Type   : 1 (Tipo de punto 1 intersección)
Curvature : ----- (Curvatura del punto)
Angle in  : ----- (Ángulo de entrada a rotonda)
Angle out : ----- (Ángulo de salida de rotonda)
/>

...

// Rotonda
Nodo N </
X      : 50.000 (Posición en eje X)
Y      : 35.000 (Posición en eje Y)
Max. Speed: 25.000 (Velocidad Máxima del tramo)
Type   : 2 (Tipo de punto 1 intersección)
Curvature : 0.100 (Inverso del radio de curvatura)
Angle in  : 25.000 (Ángulo de entrada a rotonda)
Angle out : 45.000 (Ángulo de salida de rotonda)
/>

...
```

Figura 3: Descripción de mapas por medio de puntos simples

El planificador local de trayectorias se encarga de recibir los datos suministrados por el planificador global con el objetivo de generar una ruta suave y continua en curvatura de modo que no queden comprometidas ni la seguridad ni el confort de los ocupantes del vehículo. Se utilizan para este fin curvas paramétricas, como splines o Bézier (González et al., 2014), e incluso clotoides (Alia et al., 2015). Se puede observar este comportamiento en la Figura 4. El planificador local genera un buffer de horizonte de trayectorias que contempla un número máximo de puntos en una distancia máxima (horizonte de planificación). Este buffer lo lee el planificador reactivo y puede determinar situaciones de riesgo en el entorno. El planificador reactivo ayuda a que el vehículo pueda tomar la decisión correcta a la hora de realizar cambios de carril, adelantamientos, o a evasión de obstáculos. En caso de ser necesario, realiza una modificación del buffer para poder llevar a cabo alguna de las maniobras mencionadas.

La unidad de control, con la información del horizonte de visión, realiza el seguimiento de la trayectoria mediante un controlador lateral. A su vez, las consignas de seguimiento de velocidad provienen del control longitudinal.

La unidad de control lateral genera las consignas de volante, asegurando la seguridad y la robustez ante fallos, como pueden ser la pérdida temporal de la señal GPS, o cambios de referencia inesperados. Utiliza algoritmos como por ejemplo lógica difusa (Perez et al., 2011; Hessburg and Tomizuka, 1994; Onieva et al., 2010), controladores lineales (Pérez et al., 2013; González and Pérez, 2013), controladores basados en modelos como el MPC (Falcone et al., 2007; Lee et al., 2012), entre otros.

La unidad de control longitudinal genera las señales de actuación destinadas a los pedales de acelerador y al freno, asegurando el confort de los pasajeros y proporcionando la capacidad de seguir la referencia de velocidad de forma rápida y sin cambios bruscos. Han sido utilizados algoritmos de control basados en lógica difusa como en (Milanés et al., 2009; Pérez

et al., 2011), MPC (Li et al., 2011), y controladores PID (Villagrà et al., 2010).

El bloque de actuación interpreta las consignas de control de alto nivel y genera las señales a los actuadores (acelerador, freno y volante). El control de bajo nivel recibe la consigna de posición del volante y mueve el motor de la dirección, envía las consignas a la unidad de control electrónico del acelerador y también al freno.

El IHM es el encargado de enviar parámetros de configuración a los distintos sistemas, además de monitorizar la evolución de las variables de interés y controlar el correcto funcionamiento de cada uno de los bloques. Contiene también la base de datos de los mapas que serán leídos por el subsistema de decisión, y guarda los datos generados en las pruebas.

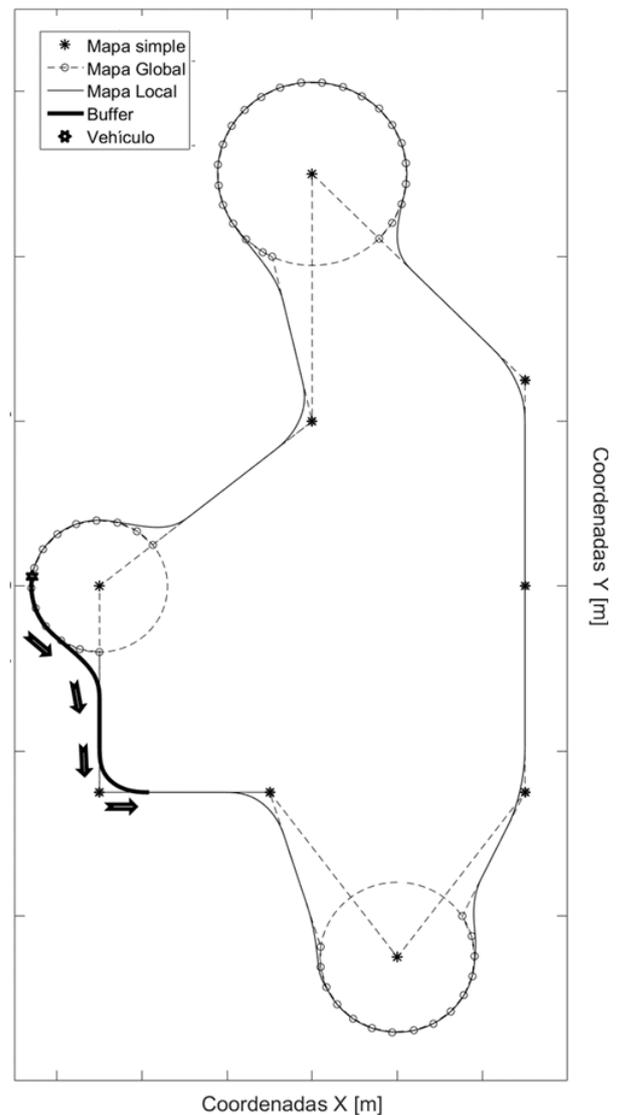


Figura 4: Etapas del módulo de decisión

4. Despliegue de la arquitectura

El despliegue de la arquitectura para la realización de pruebas virtuales y reales incluye: el simulador, el software, los vehículos de prueba y el escenario donde se llevan a cabo. A continuación, se describen cada uno de estos aspectos.

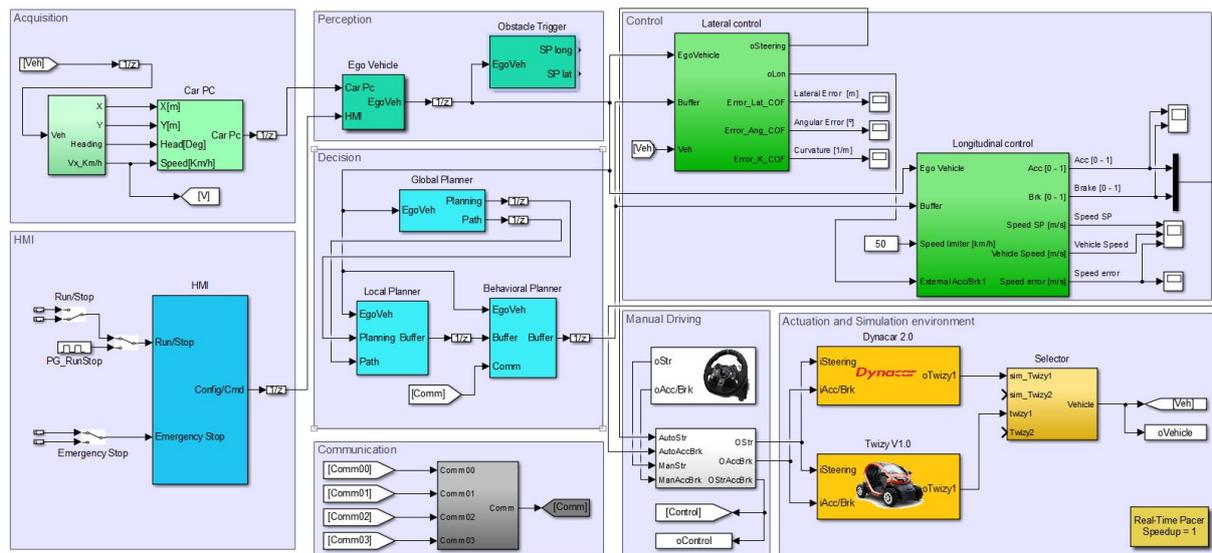


Figura 5: Arquitectura de control implementada en MATLAB[®]/Simulink[®]

4.1. Implementación de la arquitectura

La arquitectura se ha construido en torno a Matlab[®]/Simulink[®], ya que el simulador se encuentra embebido en uno de sus módulos y es necesario que el vehículo real y el virtual usen el mismo software para conducción automatizada. En la figura 5 se muestra la arquitectura implementada en Simulink[®].

Todos los bloques han sido programados en lenguaje C/C++ y embebidos en la plataforma de Simulink[®] en forma de S-functions, con el objetivo de aumentar la eficiencia en la ejecución de los algoritmos de conducción automatizada y minimizar los tiempos de migración de funciones específicas de conducción automatizada (como por ejemplo, paradas de emergencia o evasión de obstáculos), a las unidades de control electrónico (ECU, *Electronic Control Unit*) que son utilizadas en los vehículos.

4.2. Simulador: Vehículo virtual

Para una arquitectura dual es necesario garantizar que las pruebas reales y las simuladas sean lo más parecidas posible. El simulador de dinámica vehicular Dynacac[®], desarrollado por Tecnia Research & Innovation, cumple satisfactoriamente con este requisito, ya que ofrece la posibilidad de describir de forma detallada cualquier vehículo terrestre, independientemente de su tamaño o número de ejes.

4.2.1. Dynacac 2.0[®]: Modelo multi-cuerpo

El modelo de un vehículo generado por Dynacac[®] está basado en una descripción multi-cuerpo con al menos 14 grados de libertad (Pacejka, 2006) que corresponden a tres para los desplazamientos del chasis en los ejes X-Y-Z (frontal, lateral y vertical), tres para las rotaciones del chasis respecto a los ejes X-Y-Z (roll, pitch y yaw), uno por cada rueda (al menos 4) para el desplazamiento vertical, y uno por rueda para la rotación en el plano Y-Z.

El modelo realiza cálculos individuales para las ruedas (García de Jalón and Bayo, 1994), suspensiones y chasis con

la finalidad de mejorar la precisión en los cálculos y, por ejemplo, en caso de tener terrenos con tendencias poco regulares o poco planas, obtener resultados más ajustados a la realidad.

El modelo multi-cuerpo ha sido incorporado como un módulo MATLAB[®]/Simulink[®] para ser utilizado con facilidad por los usuarios del software. Algunas de las aplicaciones que se realizaron con este modelo son: pruebas de unidades hardware de control para vehículos (HiL, *Hardware in the Loop*), prueba con modelos externos (MiL, *Model in the Loop*), como por ejemplo motores y también, pruebas con conductores reales en banco de pruebas (DiL, *Driver in the Loop*) y percibiendo la respuesta del simulador (Lattarulo et al., 2017; Juez-Uriagereka et al., 2017).

4.2.2. Dynacac 2.0[®]: Visor 3D

El visor 3D de Dynacac 2.0[®] es uno de los atributos destacados con los que cuenta el vehículo virtual, ya que permite visualizar el estado del vehículo en tiempo real y tener una percepción clara del comportamiento del coche.

Permite la incorporación de varios vehículos usando el modelado multi-cuerpo, o incluso otro tipo de objetos. En las figuras 6(a) y 6(b) se muestran dos escenarios distintos en el visualizador.

4.2.3. Dynacac 2.0[®]: Editor de escenarios

El editor de escenario es otro de los módulos del software con los que cuenta la suite de Dynacac 2.0[®]. Permite generar escenarios sobre los cuales se desplazará el modelo. La descripción del escenario incluye alteraciones en el terreno, tamaños de las rutas, colocación de vegetación y objetos, entre otros aspectos. En la figura 6(c) se muestra la interfaz del software.

4.3. Vehículo real

En la actualidad, Tecnia Research and Innovation dispone de un vehículo Renault Twizy instrumentado y automatizado capaz de alcanzar 80[km/h], que se utiliza para la validación

de la arquitectura presentada. El tamaño del vehículo permite embarcar una persona y el correspondiente equipamiento de automatización (véase Figura 7).



(a) Múltiples vehículos en Dynacar



(b) Diversos tipos de objetos en Dynacar



(c) Editor de escenarios del Dynacar

Figura 6: Visualizador 3D del Dynacar y editor de escenarios

En el interior del vehículo, se dispone de un ordenador industrial, con un procesador Core™ i7 de 6ª generación (4 núcleos) que ejecuta el software Matlab®/Simulink®, conectado vía protocolo CAN con un controlador lógico programable (PLC) cuya función es realizar el control de bajo nivel de posición de los actuadores (freno y volante), así como servir de interfaz entre la electrónica embarcada y los equipos de automatización.

Los actuadores instrumentados en el vehículo son el acelerador, freno y volante. El freno se acciona mediante un actuador lineal que regula la posición del pedal. El motor de la barra de dirección lo controla el PLC utilizando la señal de posición del volante procedente del ordenador embarcado y una señal de retorno procedente del encoder de posición. La actuación del pedal acelerador se realiza por medio de una señal que se dirige a la ECU del acelerador, permitiendo el funcionamiento tradicional del vehículo usando el acelerador o de forma automática evitando el desplazamiento del pedal.

El vehículo cuenta también con un Sistema Satelital de Navegación combinado con otro de Navegación Inercial (GNSS/INS), instalado en el centro de gravedad del vehículo, y cuya función es la medición de movimiento, posición y orienta-

orientación. La antena está fijada en el techo del vehículo directamente sobre el GNSS/INS.



Figura 7: Vehículo Twizy instrumentado

Para la detección de obstáculos y verificación del entorno, cuenta con un laser de 4 capas horizontales de haces de luz, que se encuentra conectado en el frontal del vehículo a una altura de 40 centímetros del suelo. Su funcionamiento se basa en el envío de pulsos de luz para detectar objetos. A partir de las características del haz de luz recibido se puede realizar clasificación de objetos (señales de tráfico, otros vehículos, etc.).

4.4. Circuito de pruebas

Para la realización de las pruebas se dispone del circuito cerrado mostrado en la Figura 8, que permite la realización de maniobras a baja y media velocidad (40[km/h] como máximo). El circuito se encuentra en las propias instalaciones de Tecnalia en el Parque Tecnológico de Zamudio (Vizcaya, España). En él pueden simularse diversas configuraciones que posibilitan la recreación de ambientes urbanos, con carriles de pista con el ancho de vía promedio de 3 metros.



Figura 8: Circuito de pruebas

La pista posee un lugar para el asentamiento permanente de la base del GNSS/INS diferencial, permitiendo una buena cobertura.

5. Validación de la arquitectura

5.1. Escenario planteado

El escenario planteado está configurado sobre la pista de pruebas mencionada, con dos rotondas (entrada y salida tangenciales) conectadas por tramos rectos, traducándose en giros de retorno en forma de “U” (principio y final de la pista 11(a)

y 12(a)). El trazado contempla también un cambio de carril. Se muestra también el mapa con el recorrido realizado por ambos vehículos (real y virtual).

5.1.1. Validación del módulo de decisión

Para validar el módulo de decisión, se utilizaron curvas de Bézier como generadoras de trayectorias, comparándose los resultados obtenidos en las pruebas realizadas por los vehículos virtual y real en el seguimiento de las trayectorias propuestas. La Figura 9 muestra las características generales de la curva. La ecuación que define estas curvas es:

$$B(t) = \sum_{i=0}^n \binom{n}{i} (1-t)^{n-i} t^i P_i(x, y) \quad (1)$$

donde $P_i(x, y)$ son los puntos de control de la curva de Bézier, n es el orden de la curva (la curva contiene $n + 1$ puntos de control), t es el parámetro de generación de la curva, que puede tomar valores entre 0 y 1. El valor de $t = 0$ es el punto inicial de la curva que coincide con el punto de control P_0 y el valor $t = 1$ es el punto final de la curva y coincide con el punto de control P_n .

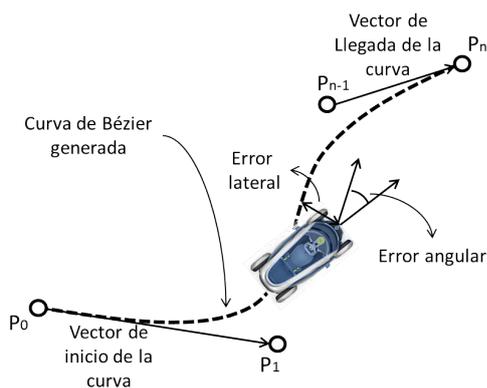


Figura 9: Curva de Bézier y variables de control

La utilización de estas curvas se debe a que cuentan con propiedades que resultan de interés para conducción automatizada tal como se presenta en (Pérez R. et al., 2014), concretamente:

- El vector de dirección de la curva de Bézier en el punto $t = 0$, es el definido por los puntos de control P_0 y P_1 y se fija para que el ángulo generado al inicio sea igual que la dirección de la vía.
- El vector de dirección en el punto $t = 1$, es el definido por los puntos de control P_{n-1} y P_n y se fija para que el ángulo generado al finalizar la trayectoria sea igual que la dirección de la vía
- Son curvas continuas, con cambios suaves de curvatura.
- Usando curvas de Bézier se puede asegurar curvatura inicial o final 0, esto es importante cuando se inicia o termina en un tramo de vía recto.

5.1.2. Validación del módulo de control

Para la validación de la arquitectura a nivel de control se plantea la utilización de dos algoritmos de control lateral. El primer controlador lleva el nombre de doble proporcional y responde a la expresión dada por la ecuación 2.

$$C_v = k_{lat} * e_{lat} + k_{ang} * e_{ang} \quad (2)$$

donde C_v es el valor de control que se encuentra saturado entre 1 y -1, e_{lat} es denominado el error de seguimiento, y es la distancia perpendicular que existe entre el centro del eje delantero y la trayectoria. El e_{ang} es el error de ángulo, y se calcula como el ángulo que se forma entre la orientación del vehículo y la orientación de la trayectoria. Las ganancias k_{lat} y k_{ang} son valores sintonizados en función del error de seguimiento y el error angular respectivamente.

El segundo algoritmo de control lateral utilizado en las pruebas de validación es una variación del anterior, que usa información acerca de la geometría de la trayectoria al incorporar la variable curvatura a modo de controlador anticipativo, por lo que se denomina controlador doble proporcional con curvatura:

$$C_v = k_{lat} * e_{lat} + k_{ang} * e_{ang} + k_{curv} * curv \quad (3)$$

donde $curv$ es el valor de la trayectoria en ese instante de tiempo y k_{curv} es la ganancia de sintonización asociada al controlador anticipativo. Este controlador es denominado controlador doble proporcional más curvatura (Pérez R. et al., 2014). En la Figura 9 se muestra el detalle y cálculo de las variables de control utilizadas.

Para el caso del control longitudinal su funcionamiento se valida mediante el comportamiento de un controlador difuso, como en (Milanés et al., 2009), para demostrar la modularidad de la arquitectura con diferentes algoritmos. Recibe como entrada el error de velocidad con respecto a la referencia, y la velocidad actual del vehículo. En la Figura 10(a) se muestra la función de pertenencia asociada al error de velocidad, mientras que en la Figura 10(b) se presenta la asociada a la velocidad. La salida del controlador se encuentra normalizada entre -1 y 1, donde el lado negativo se corresponde con la máxima posición del actuador del freno y el positivo a la máxima acción del actuador del acelerador.

Las funciones de pertenencia y las salidas del controlador difuso (Figura 10(c)) se relacionan con las reglas:

- SI **error de velocidad negativo** Y **velocidad baja** ENTONCES **Dis. L3.**
- SI **error de velocidad negativo** Y **media/baja baja** ENTONCES **Dis. L2.**
- SI **error de velocidad negativo** Y **media baja** ENTONCES **Dis. L1.**
- SI **error de velocidad centro** ENTONCES **Mantener.**
- SI **error de velocidad positivo** Y **velocidad baja** ENTONCES **Aum. L1.**
- SI **error de velocidad positivo** Y **media/baja baja** ENTONCES **Aum. L2.**
- SI **error de velocidad positivo** Y **media baja** ENTONCES **Aum. L3.**

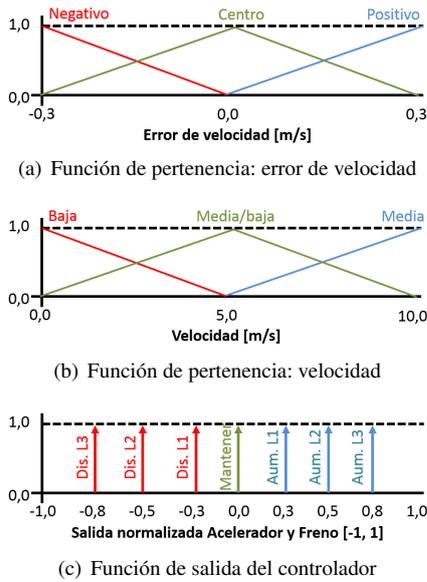


Figura 10: Parámetros del controlador difuso.

5.2. Resultados de las pruebas con el vehículo virtual

La Figura 11 muestra los resultados obtenidos para el vehículo virtual. La Figura 11(a) muestra la trayectoria generada con Bézier y el seguimiento que realiza el vehículo virtual usando el controlador doble proporcional sin curvatura, la Figura 11(b) es el error de seguimiento de la trayectoria, la Figura 11(c) es el error angular (error entre la dirección del vehículo y la dirección de la ruta) y la Figura 11(d) es el seguimiento de velocidad del vehículo y la referencia de velocidad.

La trayectoria que se muestra en la Figura 11(a) fue generada usando las dimensiones de la pista real de pruebas. Describe en su totalidad las 4 rotondas (cambios de sentido) y el cambio de carril con los que cuenta la pista. El sistema de coordenadas que se usa en este caso es el de coordenadas absolutas (Dynacar). La línea en trazo discontinuo es la trayectoria generada con Bézier y en trazo continuo y trazo con punto se muestra la posición del vehículo durante el recorrido en modo automático.

El error de seguimiento que se obtiene a partir de los dos controladores aparece reflejado en la Figura 11(b). La línea de trazo fino corresponde al controlador doble proporcional sin curvatura, y la línea de trazo con punto al controlador doble proporcional con curvatura. Los resultados numéricos quedan reflejados en la Tabla 1, pudiendo apreciarse que el error de seguimiento máximo, promedio y la mediana son mejores en el controlador que incorpora curvatura. El error de seguimiento máximo mejora en un 57 % con respecto al controlador doble proporcional, el promedio mejora en un 67 %, lo mismo que la mediana del error. En promedio, el controlador que considera la curvatura tiene un 66 % de mejoría con respecto al doble proporcional sin curvatura.

Tabla 1: Errores de seguimiento en simulación y vehículo real

Controlador	Máximo	Promedio	Mediana
Veh. virtual sin curv.	1.18 [m]	0.49 [m]	0.41 [m]
Veh. virtual con curv.	0.50 [m]	0.16 [m]	0.14 [m]
Veh. real con curv.	0.53 [m]	0.11 [m]	0.08 [m]

La Figura 11(c) muestra el error angular (error entre la dirección o ángulo del vehículo y la ruta). La línea delgada es la respuesta del controlador doble proporcional sin curvatura y la que tiene punto corresponde al controlador con curvatura. Los resultados de esta prueba se resumen en la Tabla 2. El controlador con curvatura mejoró el error máximo en 12 %, el promedio en 10 % y en 21 % la mediana. En promedio, el controlador doble proporcional con curvatura es un 14 % mejor que el doble proporcional sin curvatura.

Tabla 2: Tabla con los errores angulares en simulación y vehículo real

Controlador	Máximo	Promedio	Mediana
Veh. virtual sin curv.	25.21 [°]	7.92 [°]	7.73 [°]
Veh. virtual con curv.	22.65 [°]	7.19 [°]	6.10 [°]
Veh. real con curv.	13.10 [°]	1.96 [°]	1.49 [°]

El seguimiento de velocidad realizado por el controlador difuso se presenta en la Figura 11(d). Se aprecia que la velocidad es seguida correctamente con un ligero retraso.

5.3. Resultados de las pruebas con el vehículo real

Las pruebas con vehículo real se han realizado sobre el mismo escenario planteado para el vehículo virtual. En las pruebas de seguimiento de la trayectoria únicamente se utilizó el controlador doble proporcional con curvatura por ser el que obtuvo mejores resultados en simulación.

En la Figura 12(b) se presenta el error de seguimiento del vehículo con respecto a la ruta generada por curvas Bézier. Los valores del error máximo fueron de 0.53 m, el promedio de 0.11 m y una mediana de 0.08 m, equivalente a un error absoluto de 0.03 m, 0.05m y 0.06 m con respecto a su contraparte virtual.

El error angular se muestra en la Figura 12(c). Los valores máximo, promedio y mediana del error angular son de 13.10°, 1.96° y 1.49° respectivamente, equivalente a un error entre la simulación y la respuesta del sistema real de 9.55°, 5.23° y 4.6° en el error máximo, promedio y mediana.

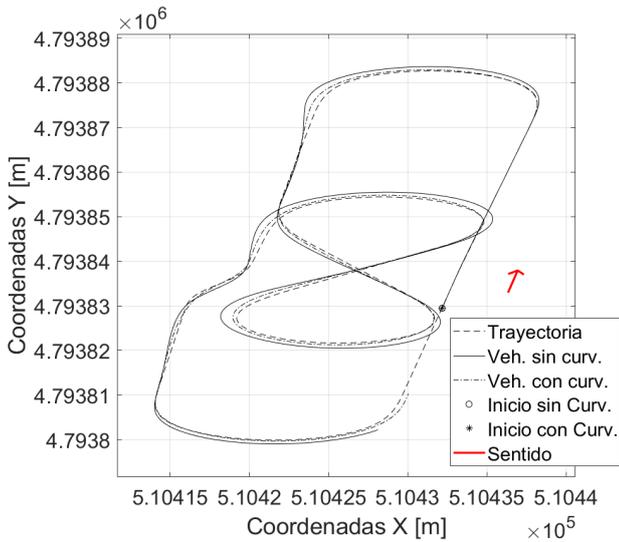
El seguimiento de velocidad realizado por el controlador difuso se presenta en la Figura 12(d), mostrando un ligero retraso de seguimiento de 0.5 segundos.

Conclusiones y trabajos futuros

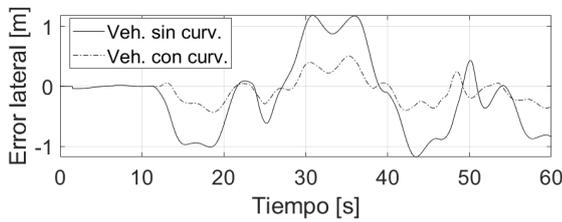
6.

En este trabajo, se presenta una arquitectura de control dual-modular para vehículos automatizados, describiendo las áreas de instrumentación del vehículo, percepción del entorno, comunicaciones, decisión y control. La mayor contribución de este trabajo es la capacidad de validar, en virtual y en real, algoritmos relacionados con la generación dinámica de trayectorias y los controladores de seguimiento.

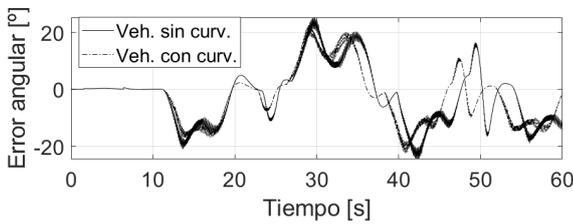
La arquitectura ha sido implementada en Matlab®/Simulink® con el objetivo de ser integrada al simulador de dinámica de vehículo Dynacar®. Los módulos y unidades de control han sido desarrollados para ser soportados por esta plataforma y en general cualquier ECU y han sido agrupadas por el tipo de tarea que desempeñan (modular), facilitando el proceso de diseño en la conducción automatizada.



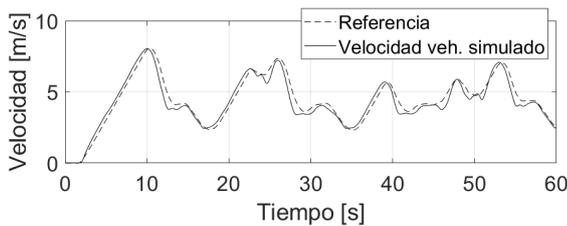
(a) Trayectoria y seguimiento con controlador sin curvatura



(b) Error de seguimiento



(c) Error angular



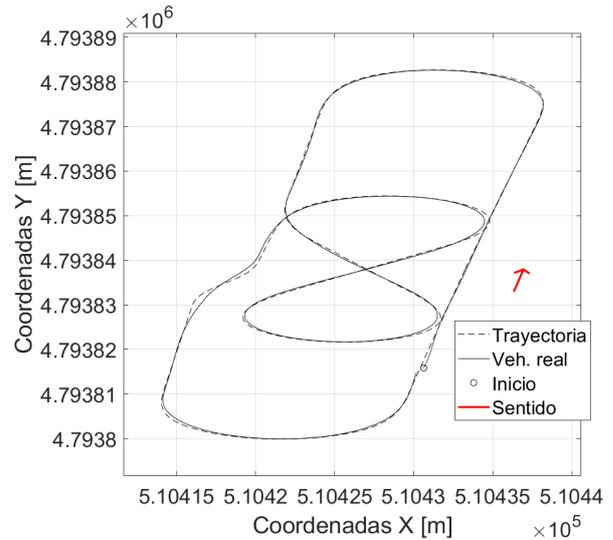
(d) Velocidad longitudinal

Figura 11: Resultados obtenidos con el vehículo virtual.

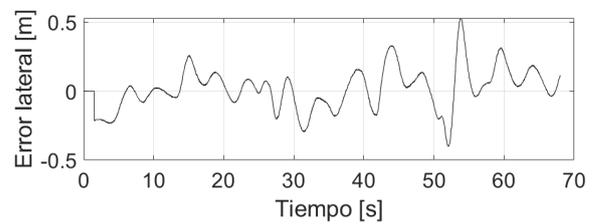
La dualidad real-virtual de la arquitectura ha sido validada con comparativas en la generación dinámica de trayectorias, seguimiento de las mismas y la velocidad. Los valores obtenidos son muy próximos entre el entorno virtual y real, debido a la precisión del modelo usado para las simulaciones (modelo multi-cuerpo). Los resultados presentan un error de menos de 10 centímetros y 10 grados entre los resultados reales y virtuales.

La arquitectura ha sido planteada de forma genérica, para poder contemplar cualquier escenario durante el proceso de automatización de un vehículo, además ha permitido realizar

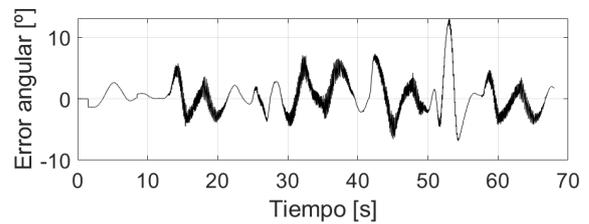
pruebas exitosas en: generación de trayectorias incluyendo perfil de velocidad usando control predictivo basado en modelo (Lattarulo et al., 2018c), generación de perfiles de velocidad con curvas paramétricas (Lattarulo et al., 2018d), comparativas de controladores longitudinales (Marcano et al., 2018), estudios de algoritmos de evasión basados en curvas de Bézier (Lattarulo et al., 2018a), verificación formal de control basado en modelos para vehículos automatizados (Lattarulo et al., 2018b), diseño de un controlador longitudinal usando MPC (Matute et al., 2018), entre otros.



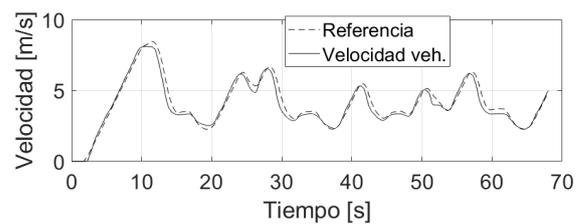
(a) Trayectoria



(b) Error de seguimiento



(c) Error angular



(d) Velocidad longitudinal

Figura 12: Resultados del vehículo real.

Las líneas futuras de investigación contemplan la verificación y validación de maniobras complejas con la incorporación a carril y maniobras cooperativas.

Agradecimientos

Los autores desean agradecer al proyecto ECSEL ENABLE-S3 con el número 692455-2 por el soporte en el desarrollo del presente trabajo.

Referencias

- Alia, C., Gilles, T., Reine, T., Ali, C., 2015. Local trajectory planning and tracking of autonomous vehicles, using clothoid tentacles method, in: 2015 IEEE Intelligent Vehicles Symposium (IV), pp. 674–679.
- Bagheri, M., Siekkinen, M., Nurminen, J.K., 2014. Cellular-based vehicle to pedestrian (V2p) adaptive communication for collision avoidance, in: 2014 International Conference on Connected Vehicles and Expo (ICCVE), pp. 450–456.
- Bengler, K., Dietmayer, K., Farber, B., Maurer, M., Stiller, C., Winner, H., 2014. Three Decades of Driver Assistance Systems: Review and Future Perspectives. *IEEE Intelligent Transportation Systems Magazine* 6, 6–22.
- Berntorp, K., Hoang, T., Quirynen, R., Cairano, S.D., 2018. Control architecture design for autonomous vehicles. *Conference on Control Technology and Applications (CCTA)*.
- Bertoncello, M., Wee, D., 2015. Ten ways autonomous driving could redefine the automotive world | McKinsey & Company.
- Falcone, P., Borrelli, F., Asgari, J., Tseng, H.E., Hrovat, D., 2007. Predictive Active Steering Control for Autonomous Vehicle Systems. *IEEE Transactions on Control Systems Technology* 15, 566–580.
- Favará, F.M., Nader, N., Eurich, S.O., Tripp, M., Varadaraju, N., 2017. Examining accident reports involving autonomous vehicles in California. *PLoS one* 12(9).
- González, D., Pérez, J., 2013. Control architecture for Cybernetic Transportation Systems in urban environments, in: 2013 IEEE Intelligent Vehicles Symposium (IV), pp. 1119–1124.
- González, D., Pérez, J., Lattarulo, R., Milanés, V., Nashashibi, F., 2014. Continuous curvature planning with obstacle avoidance capabilities in urban scenarios, in: 2014 IEEE 17th International Conference on Intelligent Transportation Systems (ITSC). IEEE, Qingdao, China.
- González, D., Pérez, J., Milanés, V., Nashashibi, F., 2016. A review of motion planning techniques for automated vehicles. *IEEE Transactions on Intelligent Transportation Systems*.
- Harding, J., Powell, G., Yoon, R., Fikentscher, J., Doyle, C., Sade, D., Lukuc, M., Simons, J., Wang, J., 2014. Vehicle-to-Vehicle Communications: Readiness of V2V Technology for Application URL: <https://trid.trb.org/view.aspx?id=1323282>.
- Hessburg, T., Tomizuka, M., 1994. Fuzzy logic control for lateral vehicle guidance. *IEEE Control Systems* 14, 55–63.
- García de Jalón, J., Bayo, E., 1994. Kinematic and Dynamic Simulation of Multibody Systems. The Real-Time Challenge. Springer-Verlag New York. Springer-Verlag New York ed., Springer-Verlag New York.
- Jochem, T., Pomerleau, D., Kumar, B., Armstrong, J., 1995. PANS: a portable navigation platform, *IEEE*. pp. 107–112. URL: <http://ieeexplore.ieee.org/document/528266/>.
- Jones, T., Lennox, S., Sgueglia, J., Demerly, J., Zervoglos, N.A., Yang, H.H., 2018. Autonomous vehicle: modular architecture.
- Juez-Uriagereka, G., Lattarulo, R., Perez Rastelli, J., Amparan Calonge, E., Ruiz Lopez, A., Espinoza Ortiz, H., 2017. Fault Injection method for Safety and Controllability Evaluation of Automated Driving, in: Fault Injection method for Safety and Controllability Evaluation of Automated Driving, Redondo Beach, California. pp. 1867 – 1872.
- Klaus, T.C., Twitty, C.K., ERLIEN, S.M., Kegelman, J.C., Price, C.A., SCHUH, A.B., SILVERMAN, B.J., SWITKES, J.P., 2018. Automated vehicle control system architecture.
- Kress-Gazit, H., Pappas, G.J., 2008. Automatically synthesizing a planning and control subsystem for the DARPA urban challenge, in: 2008 IEEE International Conference on Automation Science and Engineering, pp. 766–771.
- Lattarulo, R., González, L., Martí, E., Matute, J., Marcano, M., Pérez, J., 2018a. Urban motion planning framework based on n-bézier curves considering comfort and safety. *Journal of Advanced Transportation*.
- Lattarulo, R., Hess, D., Matute, J.A., Pérez, J., 2018b. Towards conformant models of automated electric vehicles. *IEEE International Conference on Vehicular Electronics and Safety*.
- Lattarulo, R., Hess, D., Pérez, J., 2018c. A linear model predictive planning approach for overtaking manoeuvres under possible collision circumstances. *IEEE Intelligent Vehicles Symposium (IV)*, 1340 – 1345.
- Lattarulo, R., Martí, E., Marcano, M., Matute, J., Pérez, J., 2018d. A speed planner approach based on bézier curves using vehicle dynamic constraints and passengers comfort. *IEEE International Symposium on Circuits and Systems (ISCAS)*, 1 – 5.
- Lattarulo, R., Pérez, J., Dendaluce, M., 2017. A complete framework for developing and testing automated driving controllers, in: A complete framework for developing and testing automated driving controllers, IFAC, Toulouse, France.
- Lee, S.H., Lee, Y.O., Kim, B.A., Chung, C.C., 2012. Proximate model predictive control strategy for autonomous vehicle lateral control, in: 2012 American Control Conference (ACC), pp. 3605–3610.
- Li, S., Li, K., Rajamani, R., Wang, J., 2011. Model Predictive Multi-Objective Vehicular Adaptive Cruise Control. *IEEE Transactions on Control Systems Technology* 19, 556–566.
- Marcano, M., Matute, J.A., Lattarulo, R., Martí, E., Pérez, J., 2018. Low speed longitudinal control algorithms for automated vehicles in simulation and real platforms. *Hindawi Complexity*.
- Matute, J.A., Marcano, M., Asier Zubizarreta, J.P., 2018. Longitudinal model predictive control with comfortable speed planner. *IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*.
- Milanés, V., Onieva, E., Pérez, J., de Pedro, T., González, C., 2009. Control de Velocidad basado en Lógica Borrosa para Entornos Urbanos Congestionados. *Revista Iberoamericana de Automática e Informática Industrial RIAI* 6, 61–68.
- Milanes, V., Villagra, J., Godoy, J., Simo, J., Perez, J., Onieva, E., 2012a. An Intelligent V2i-Based Traffic Management System. *IEEE Transactions on Intelligent Transportation Systems* 13, 49–58.
- Milanes, V., Villagra, J., Perez, J., Gonzalez, C., 2012b. Low-Speed Longitudinal Controllers for Mass-Produced Cars: A Comparative Study. *IEEE Transactions on Industrial Electronics* 59.
- Onieva, E., Milanés, V., Pérez, J., de Pedro, T., 2010. Estimación de un Control Lateral Difuso de Vehículos. *Revista Iberoamericana de Automática e Informática Industrial RIAI* 7, 91–98.
- Ozguner, U., Redmill, K.A., Broggi, A., 2004. Team TerraMax and the DARPA grand challenge: a general overview, in: *IEEE Intelligent Vehicles Symposium, 2004*, pp. 232–237.
- Pacejka, H.B., 2006. Tire and Vehicle Dynamics, in: Pacejka, H.B. (Ed.), *Tire and Vehicle Dynamics*. 2 ed., Butterworth-Heinemann, Oxford, p. 642.
- Perez, J., Milanes, V., Onieva, E., 2011. Cascade Architecture for Lateral Control in Autonomous Vehicles. *IEEE Transactions on Intelligent Transportation Systems* 12, 73–82.
- Pérez, J., Milanés, V., Onieva, E., Godoy, J., Alonso, J., 2011. Longitudinal fuzzy control for autonomous overtaking, in: 2011 IEEE International Conference on Mechatronics, pp. 188–193.
- Pérez, J., Nashashibi, F., Lefaudeux, B., Resende, P., Pollard, E., 2013. Autonomous Docking Based on Infrared System for Electric Vehicle Charging in Urban Areas. *Sensors (Basel, Switzerland)* 13, 2645–2663.
- Pérez R., J., Lattarulo, R., Nashashibi, F., 2014. Dynamic trajectory generation using continuous-curvature algorithms for door to door assistance vehicles, in: 2014 IEEE Intelligent Vehicles Symposium Proceedings, pp. 510–515.
- Tas, Ö.S., Hörmann, S., Schäufele, B., Kuhn, F., 2018. Automated vehicle system architecture with performance assessment. *IEEE International Conference on Intelligent Transportation Systems*.
- The International Traffic Safety Data and Analysis Group IRTAD, 2017. IRTAD: Road Safety Annual Report 2016. URL: <https://www.itf-oecd.org/road-safety-annual-report-2016>.
- Thrun, S., 2006. Winning the DARPA Grand Challenge: A Robot Race through the Mojave Desert, in: 21st IEEE/ACM International Conference on Automated Software Engineering (ASE'06).
- Villagrà, J., Milanés, V., Pérez, J., de Pedro, T., 2010. Control basado en PID inteligentes: aplicación al control de crucero de un vehículo a bajas velocidades. *Revista Iberoamericana de Automática e Informática Industrial RIAI* 7, 44–52.
- Woo, H.J., Park, S.B., Kim, J.H., 2008. Research of the optimal path planning methods for unmanned ground vehicle in DARPA Urban Challenge, in: 2008 International Conference on Control, Automation and Systems, pp. 586–589.
- Yoon, J., Crane, C.D., 2008. LADAR based obstacle detection in an urban environment and its application in the DARPA Urban challenge, in: 2008 International Conference on Control, Automation and Systems, pp. 581–585.