

Maniobras cooperativas aplicadas a vehículos automatizados en entornos virtuales y reales

C.E. Hidalgo^{a,c,*}, M. Marcano^{b,c}, G. Fernández^a, J. M. Pérez^c

^aUniversidad Simón Bolívar, Caracas, Distrito Capital, Venezuela.

^bUniversidad del País Vasco, Bilbao, Vizcaya, España.

^cTecnalia Research & Innovation, Derio, Vizcaya, España.

Resumen

En los últimos años los Sistemas Inteligentes de Transporte, ITS (del inglés, Intelligent Transportation System) se han convertido en una realidad dentro de la sociedad, aportando soluciones y beneficios a la conducción. Con el fin de contribuir a su desarrollo, el presente trabajo describe un marco cooperativo híbrido capaz de validar maniobras entre múltiples vehículos (virtuales y reales), con el fin de disminuir los costos, tiempos y riesgos asociados al ajuste de los controladores. Para su validación se presentan 3 casos de estudio. El primero consiste en utilizar dos vehículos virtuales para realizar un Control de Crucero Adaptativo, ACC (del inglés, Adaptive Cruise Control) con seguidor de trayectoria. El segundo, emplea un coche real como seguidor y un coche virtual como líder para la maniobra de Stop & Go. Finalmente, se utilizan dos vehículos reales para el ACC. Los algoritmos de seguimiento empleados para las maniobras cooperativas están basados en controladores de lógica borrosa. Los resultados demuestran la versatilidad del marco propuesto, al poder ejecutar las maniobras correctamente en cada uno de los entornos.

Palabras Clave:

Maniobras Cooperativas, Marco Cooperativo Híbrido, ACC, Stop & Go, ITS, Lógica Borrosa.

Cooperative maneuvers applied to automated vehicles in real and virtual environments

Abstract

In recent years, Intelligent Transportation Systems (ITS) have become a reality within society, by providing benefits and solutions to the conduction. With the aim of contributing with the development of the ITS, the present work describes a hybrid cooperative framework for the validation of maneuvers between multiple vehicles (virtual and real), in order to reduce cost, time and risks associated with the controllers adjustment. For its validation three case of studies are presented. The first one consists of using two virtual vehicles to perform an Adaptive Cruise Control (ACC) with trajectory tracker. The second one, in using a real car as the follower and a virtual vehicle as the lider to perform a Stop & Go. And finally, two real cars are used to carry out an ACC. The tracker algorithms employed for the cooperative maneuvers are based in fuzzy logic controllers. The results show the versatility of the proposed framework, which was able to correctly execute the maneuvers in each of the test environments.

Keywords:

Cooperative Maneuvers, Hybrid Cooperative Framework, ACC, Stop & Go, ITS, Fuzzy Logic.

1. Introducción

Los Sistemas Inteligentes de Transporte, ITS (del inglés, Intelligent Transportation System) hacen uso de distintas tecnologías para la recolección de información y la conexión con otros agentes que conforman la red de transporte, con el fin de permitir una conducción segura, cómoda y eficiente (Fernandez

and Ito, 2016). Dentro del campo de los ITS destacan las maniobras cooperativas, las cuales surgen como una opción viable para obtener confiabilidad al momento de conducir. Distintas maniobras han sido desarrolladas e implementadas en la actualidad, entre ellas: adelantamientos en carreteras de doble sentido (Pérez et al., 2010), intersecciones (Zohdy et al., 2012), Con-

* Autor para correspondencia: carlos.hidalgo@tecnalia.com

trol de Crucero Adaptativo, ACC (del inglés, Adaptive Cruise Control) (Schmied et al., 2016), Stop & Go (Marzbanrad et al., 2015), entre otras. De las cuales se pueden destacar las últimas dos, debido a la dificultad de adaptar los sistemas de control a situaciones donde se requieren bajas velocidades, el cual es el objetivo de las maniobras presentadas en este trabajo.

En las maniobras cooperativas las acciones no dependen únicamente del propio vehículo, sino de otras variables, como por ejemplo: el comportamiento de los otros coches presentes en el entorno, las condiciones del ambiente, alta congestión de vehículos, entre otras (During and Lemmer, 2016). Debido a la diferencia que existe entre los entornos de simulación y los escenarios reales, resulta complejo poder llevar a cabo una completa extrapolación de la arquitectura de un entorno a otro, por lo cual la validación de las maniobras se puede ver comprometida. Es por esto, que muchos estudios se suelen centrar en un solo escenario, ya sea virtual o real. Encontrándose en esta problemática una de las principales contribuciones del presente trabajo, la cual consiste en la combinación de ambos entornos de prueba para la validación de maniobras cooperativas.

Por otra parte, diversos autores han implementado distintas estrategias de control para las maniobras ACC y Stop & Go. Entre estas estrategias destaca el uso de la lógica borrosa, la cual surge de la búsqueda en describir sistemas complejos mediante reglas lingüísticas (Sugeno, 1985). Los controladores difusos permiten introducir un enfoque humano al diseño de controladores, no requiriendo del modelo matemático de los métodos de diseño convencionales (Driankov et al., 1996).

Villagrà et al. (Villagrà et al., 2010) presentan una comparativa entre un controlador borroso y un PID inteligente para la implementación de la maniobra Stop & Go, empleando como entradas los errores de velocidad y distancia. Ambas estrategias presentaron un desempeño muy similar, en términos de velocidad y distancia intervehicular. Para la validación de los algoritmos se usaron sólo plataformas virtuales. En (Milanés et al., 2012) se encuentra otro caso de comparación de estrategias de control. Este trabajo evalúa el desempeño de un controlador borroso y un PI inteligente, haciendo uso de las mismas entradas que en el estudio anterior, con la diferencia que en este artículo se realizaron las pruebas en simulación, para luego validarlas en vehículos reales. Como resultados, obtuvieron de igual forma un comportamiento muy similar, con la salvedad que para el entorno real el controlador borroso presentó un mejor desempeño, al adaptarse de adecuadamente a las circunstancias presentes en dicho entorno.

En relación con la maniobra de ACC, distintos enfoques relacionados con la lógica borrosa han sido implementados. En (Rizvi et al., 2014) realizan un controlador borroso que posee como entradas, la distancia entre vehículos, la velocidad, la variación de la velocidad, y por último la velocidad relativa con respecto al coche líder, y como salida la señal de control sobre el acelerador y freno, mientras que en (Jena et al., 2016) utilizan como entradas el error en la velocidad, la variación de dicho error y la aceleración del coche, teniendo como salida de igual forma la señal sobre los actuadores. En ambos casos se presentan resultados en simulación, demostrando la versatilidad y eficacia de este tipo de controlador sobre los convencionales.

Las maniobras mencionadas anteriormente ejercen acción sólo sobre el mecanismo de control longitudinal de los vehícu-

los. Por otra parte, la mayoría de los sistemas enfocados en control lateral toman en consideración una trayectoria ya predefinida. Ejemplos de estos sistemas de control pueden ser encontrados en (Wang et al., 2015) y (Rastelli and Peñas, 2015), en donde utilizan controladores basados en lógica borrosa, que reciben como entradas el error lateral y angular del vehículo con respecto a ruta. Sin embargo, es poco usual que se use la trayectoria que pueda generar otro vehículo, con el que se mantenga un intercambio de datos, lo cual permitiría, junto al ACC, obtener un seguimiento completo del coche comunicado.

Por otra parte, este tipo de maniobras requiere que los vehículos se encuentren interconectados, para lo cual se emplean sistemas de comunicación vehiculares (Festag, 2014), que se pueden clasificar en distintos tipos: vehículos con vehículos, V2V (del inglés, Vehicle to Vehicle), vehículos con infraestructura, V2I (del inglés, Vehicle to Infrastructure) o una combinación de los distintos integrantes de los ITS, V2X (del inglés, Vehicle to Everything) (Da Cunha et al., 2014). Dichos sistemas, aportan versatilidad y efectividad a la hora de ejecutar las maniobras, al poseer distintas características, como lo pueden ser: alta velocidad en el intercambio de datos, alto rango de alcance, entre otras.

En este trabajo se presenta un marco cooperativo híbrido capaz de validar maniobras con dos vehículos en tres entornos de pruebas distintos: 1) *Virtual*, empleando el simulador Dynacar, 2) *Real*, haciendo uso de dos Renault Twizy y, 3) *Mixto*, el cual involucra un vehículo simulado como líder y uno real como seguidor. Más concretamente las maniobras desarrolladas son, ACC, Stop & Go y Control Lateral, para las cuales se utilizaron comunicaciones V2V, en donde ambos vehículos intercambian información referente a su posición, velocidad y demás estados del sistema.

El resto del artículo se divide de la siguiente manera: en la sección 2 se explican las plataformas de prueba, la arquitectura de conducción automatizada y los entornos de prueba usados para la validación de las maniobras. En la sección 3 se describen las maniobras y los algoritmos de control. En la sección 4 se describe el entorno donde se realizaron las pruebas, así como los resultados obtenidos, y finalmente en la sección 5, se presentan las conclusiones, además de los trabajos futuros que se puedan desarrollar.

2. Plataformas y Marco Cooperativo

Esta sección presenta una descripción de las plataformas utilizadas en este estudio: 1) la herramienta de simulación Dynacar (Lattarulo et al., 2017b) y 2) vehículo Renault Twizy automatizado. Ambos utilizan la arquitectura de control generalizada para la conducción automatizada presentada en (González et al., 2016). Al finalizar se detallan los entornos de pruebas basados en el marco cooperativo.

2.1. Plataforma Virtual

Dynacar es una herramienta de simulación elaborada por Tecnia Research & Innovation, que permite diseñar y probar distintos modelos de coches en ambientes completos y personalizables. Dicho simulador consta de dos entornos: 1) el modelo del vehículo en la plataforma Simulink, utilizando el modelo multi-cuerpo presentado en (Cuadrado et al., 2013) para

obtener los estados del coche y 2) el Visor 3D (Figura 1), que permite visualizar el comportamiento del vehículo en distintos escenarios.



Figura 1: Visor 3D

2.2. Plataforma Real

La plataforma experimental utilizada es un Renault Twizy 80, un vehículo eléctrico capaz de alcanzar una velocidad máxima de 80 km/h, el cual ha sido instrumentado para que el control de los actuadores sea realizado a través de una red de bus CAN. Esta plataforma se puede dividir en tres sistemas principales manejados por diferentes dispositivos, dichos sistemas son: 1) Volante: dirigido por un motor de paso basado en señales PWM, 2) Acelerador: controlado por la unidad de control del motor (ECU), que recibe una señal de voltaje analógica de 0-10 VDC, 3) Freno: manejado por un actuador mecánico lineal (Marcano et al., 2018).

2.3. Arquitectura de Control

La arquitectura de control utilizada en cada una de las plataformas consta de 6 bloques: Adquisición, Percepción, Comunicación, Decisión, Control y Actuadores. Dicha arquitectura ha sido empleada en distintos trabajos previos, de los cuales se pueden resaltar: la validación de un control lateral (Sriranjan et al., 2017) y de distintos controladores longitudinales (Marcano et al., 2018), así como también el desarrollo de un maniobra cooperativa de adelantamiento (Lattarulo et al., 2017a). Una breve descripción de cada módulo es dada a continuación:

- **Adquisición:** responsable de recolectar las señales provenientes de los sensores incorporados en el vehículo real (sensores de visión, GPS diferencial, etc), así como también del modelo multi-cuerpo presente en Dynacar.
- **Percepción:** bloque encargado de recolectar y organizar los datos provenientes del módulo de adquisición para generar un entorno estructurado que permita la movilización del vehículo de manera segura.
- **Comunicación:** se encarga de llevar a cabo la conexión entre vehículos, ya sean reales, virtuales o una combinación de ambos. Para comunicar los coches se emplean las redes W-Lan, las cuales bajo ciertas condiciones, que son mostradas en (Lin et al., 2010) y (Milanes et al., 2012) pueden emular una Red Vehicular Ad-Hoc (VANET). Además, incorpora un sub-módulo para monitorear el vehículo con el cual se está realizando la comunicación con el Visor 3D.
- **Decisión:** módulo que genera la trayectoria seguida por el vehículo. Para obtener la ruta el módulo se divide en dos sub bloques: 1) el planificador global, responsable de generar la trayectoria, y 2) el planificador local, encargado de suavizar

dicha trayectoria a través de distintos métodos como lo pueden ser curvas de Bezier (Lattarulo et al., 2018).

- **Control:** bloque encargado de generar las señales de control, longitudinal (acelerador y freno) y lateral (volante). Para cumplir este objetivo, el módulo recibe la información del bloque de decisión y ejecuta las acciones necesarias. Las señales que se obtienen se presentan de forma normalizada.
- **Actuación:** se encarga de traducir las salidas obtenidas en el bloque de control, a señales de acción real sobre el acelerador, freno y volante. Dichas señales son posteriormente enviadas a los respectivos módulos del vehículo real y virtual.

2.4. Entornos de Prueba

Para la validación de las maniobras cooperativas presentadas, se establecieron tres entornos de pruebas: 1) *Virtual*, empleando Dynacar, 2) *Real*, utilizando dos Renault Twizy y 3) *Mixto*, usando un vehículo simulado como líder de la maniobra y un vehículo real como seguidor.

2.4.1. Entorno Virtual

Se emplean dos ordenadores conectados a una misma red W-Lan, en los cuales se pone en marcha el simulador Dynacar, junto con el Visor 3D, donde pueden ser observados los vehículos que están realizando las maniobras (Figura 2).



(a) Simulador del ordenador correspondiente al vehículo seguidor



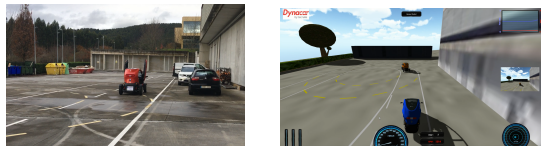
(b) Simulador del ordenador correspondiente al vehículo líder

Figura 2: Vista del simulador en ambos ordenadores

La validación de maniobras en simulación permite que el desarrollo y ajuste de los controladores, se haga de una manera segura y eficiente, adicionalmente hace posible una efectiva corrección y detección de errores. Sin embargo, el comportamiento de los vehículos reales se ve afectado por distintas variables que no se consideran en el ámbito virtual, como por ejemplo: el modelado de los actuadores y el tiempo en hacerse efectivo el comando de control enviado desde el ordenador central, debido a retrasos en los sistemas de PLC y del controlador del motor. Debido a esto, los resultados obtenidos haciendo uso de este entorno sirven como guía para entender el funcionamiento de los coches en las maniobras, más no definen correctamente su comportamiento real.

2.4.2. Entorno Mixto

Se hace uso de un vehículo real (Twizy), conectado mediante una red W-Lan a un ordenador, el cual simula un coche virtual a través del simulador Dynacar (Figura 3). Además, se puede monitorear el comportamiento del vehículo real mediante el Visor 3D.



(a) Vehículo real realizando una maniobra junto a un vehículo virtual (b) Vehículo virtual y recibiendo información del vehículo real

Figura 3: Ejemplo de un entorno mixto

En este caso se pueden obtener los resultados en el ámbito real, teniendo en cuenta las variables que puedan afectar los vehículos, de forma segura, ya que el coche con el cual se está realizando la maniobra es virtual. Adicionalmente, se reducen los tiempos de ensayos y se aumenta la efectividad en la depuración de los sistemas. Siendo así, el entorno con mayor cantidad de beneficios a la hora de validar maniobras.

2.4.3. Entorno Real

Se emplean dos coches Twizy conectados de igual forma por una red W-Lan, en donde se puede probar el funcionamiento de los controladores de las maniobras en el escenario real (Figura 4).



Figura 4: Vehículos reales realizando un ACC.

Utilizando este entorno se obtiene la respuesta completa de los vehículos al realizar la maniobra, tomando en consideración todas las variables externas que se puedan presentar. Si bien el resultado conseguido de esta forma valida en su totalidad el controlador para la maniobra cooperativa, poder efectuar este tipo de pruebas requiere de un mínimo de plataformas experimentales que involucran una gran cantidad de recursos y tiempo, además de presentar mayor riesgo de siniestros al momento de ocurrir una falla.

Para los entornos descritos, se siguió el esquema planteado en la Figura 5, donde primero se reúne la información proveniente del modelo multi-cuerpo (Dynacar) o de los sensores (Twizy), luego se organiza y se envía al otro vehículo, ya sea virtual o real. Con respecto al entorno real, el coche encargado de realizar las maniobras procesa los datos y realiza las acciones de control necesarias. En el caso virtual, ambos ordenadores envían los datos al Visor 3D para poder monitorear los vehículos con los cuales se realiza la comunicación. En el entorno mixto el vehículo virtual cumple el rol de líder, por lo que se encarga de enviar información al otro coche y al Visor 3D, en donde se observa el coche real. Este vehículo cumple la función del seguidor, por lo que se encarga de realizar las maniobras y enviar sus datos para ser monitorizado.

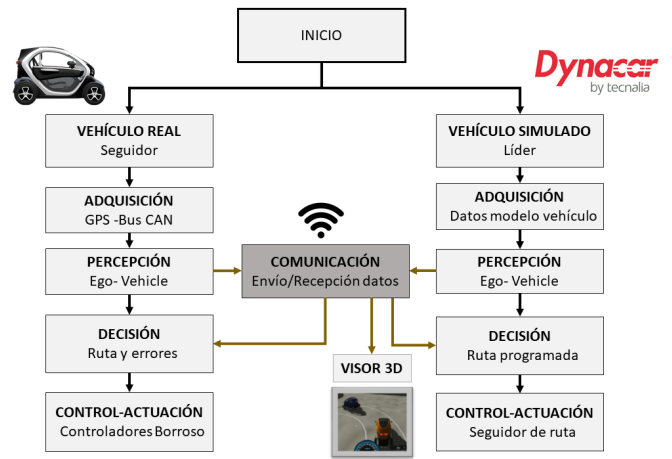


Figura 5: Arquitectura lógica de la implementación de la maniobra

3. Diseño de las Maniobras

En esta sección se expone una descripción de los controladores basados en lógica borrosa, así como el desarrollo de las maniobras cooperativas de ACC, Stop & Go y Control Lateral, para las cuales, primero se detalla de forma breve en que consisten, para luego ahondar en el algoritmo de control empleado y su debida implementación.

Los controladores borrosos permiten dar valores intermedios entre dos extremos a una variable que se desea manipular, con lo cual se hace posible que la lógica humana sea empleada de manera directa. Están compuestos de tres etapas, *Fuzzyficación*, *Método de Inferencia* y *Desfuzzysificación*, además de una *Base de Reglas*. En la Figura 6 se puede encontrar un esquema de la representación de los controladores borrosos, donde cada etapa consiste en:

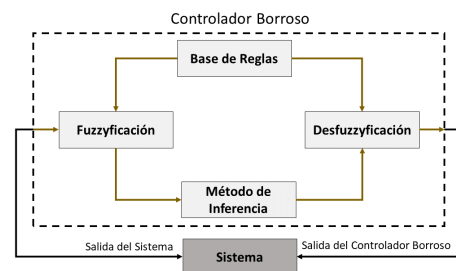


Figura 6: Esquema de un controlador borroso.

- **Fuzzyficación:** se encarga en transformar la(s) variable(s) de entrada en un grado de pertenencia. Para ejecutar este proceso se utilizan las funciones de pertenencia (comúnmente poseen formas de triángulos, trapezoides, gaussianas, etc.) con las cuales definen el grado de pertenencia de un conjunto a otro.
- **Método de Inferencia:** es la etapa donde se evalúan las entradas fuzzyficadas a través de condiciones definidas previamente. En la actualidad existen varios métodos para realizar esta labor, destacando de entre ellos el método de inferencia de Mamdani (Iancu, 2012). Este método es el utilizado por los controladores descritos en el presente trabajo.

- **Base de Reglas:** conjunto de reglas lógicas de la forma *IF-THEN*, que aplican la lógica humana para definir las salidas del controlador.
- **Desfuzzyficación:** etapa donde se establecen las salidas del controlador a partir de las reglas evaluadas anteriormente. De igual forma, existen distintos métodos para obtener dicha salida, siendo el más relevante, el método del centroide (Morcillo, 2011).

3.1. ACC y Stop & Go

La maniobra de ACC permite fijar una velocidad de conducción y mantenerla de forma automática, guardando cierta distancia de seguridad con respecto al vehículo delantero, actuando sobre el acelerador y freno del coche. Mientras que la maniobra de Stop & Go busca evaluar la capacidad del vehículo para acelerar y desacelerar en situaciones de tráfico, o frenados de emergencia producidos por el coche líder.

3.1.1. Algoritmo de control

En ambas maniobras se empleó el mismo controlador basado en lógica difusa, para el cual se hizo uso del Toolbox de MATLAB para controladores borrosos. Las entradas de control corresponden al error de distancia y velocidad entre ambos vehículos para mantener la posición del coche a una distancia de referencia previamente establecida, basada en parámetros de seguridad.

En la Figura 7 se presenta el escenario descrito, donde se pueden observar tres distancias d_{l-s} , d_{ref} y d_s . Siendo d_{l-s} la distancia euclidiana entre la zona frontal del vehículo seguidor y la parte trasera del líder; d_{ref} es la distancia de referencia establecida y d_s la distancia de seguridad, la cual a efectos del presente trabajo posee un valor de 7 m, fijado experimentalmente.

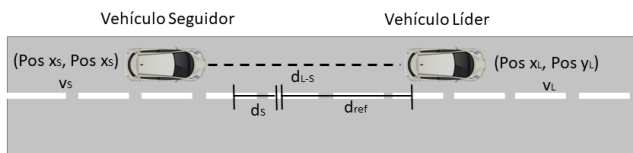


Figura 7: Características del CACC implementado.

Considerando estas variables, se realiza el cálculo de los errores basado en la posición de los vehículos y sus respectivas velocidades. El *error de distancia* (e_{dist}) se calcula como la diferencia de la distancia real entre los vehículos, d_{l-s} y la distancia de referencia, d_{ref} (Ecuación 1).

$$e_{dist} = d_{l-s} - d_{ref} \tag{1}$$

Donde la separación entre los vehículos es calculada como la distancia Euclidiana entre dos puntos (Ecuación 2), más específicamente los puntos correspondientes a las posiciones de los coches ((x_l, y_l)), para el vehículo líder, y (x_s, y_s) para el vehículo seguidor).

$$d_{l-s} = \sqrt{(x_l - x_s)^2 + (y_l - y_s)^2} \tag{2}$$

En cuanto a la distancia de referencia, la misma es calculada de manera discreta a través de la fórmula del cuadrado de la

velocidad, en este caso del vehículo seguidor (v_s), dada por la Dirección General de Tráfico Española, DGT (Española, 2015), complementada con una distancia de seguridad (d_s) (Ecuación 3).

$$d_{ref} = \left[\frac{v_s}{10}\right]^2 + d_s \tag{3}$$

En lo que respecta al *error de velocidad* (e_{vel}), el mismo se calcula como la diferencia entre la velocidad del vehículo líder (v_l) y la del coche seguidor (v_s) (Eq. 4).

$$e_{vel} = v_l - v_s \tag{4}$$

3.1.2. Diseño del controlador borroso

Para el *error de distancia* se definieron cuatro funciones de pertenencia (Figura 8), tres para mantener el error cerca de la referencia, que en este caso es fijada en el rango de [-2;1] metros y una para el caso en que la distancia sea muy positiva, para lo cual se incrementa la acción sobre el acelerador.

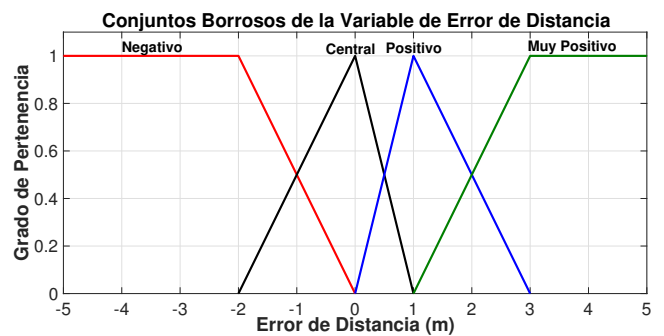


Figura 8: Función de pertenencia del error de la distancia.

En el caso del *error de velocidad* se definieron cuatro funciones de pertenencia (Figura 9), tres de ellas en el intervalo [-1;1.5] km/h, para mantener la velocidad lo más cercano posible a la del vehículo líder, y una para los valores muy negativos, con la cual se busca ejercer mayor acción sobre el freno.

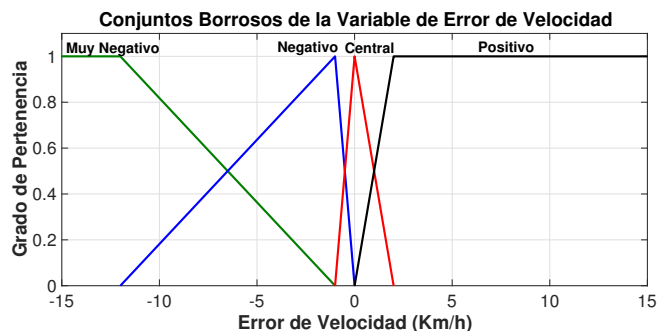


Figura 9: Función de pertenencia del error de la velocidad.

La salida del controlador corresponde a la señal normalizada sobre los actuadores, para la cual se definieron cinco valores del tipo *singleton* (Sugeno and Murakami, 1984), dentro del conjunto [-1;1]. De estos valores, los que se encuentran en el rango de [-1;0] pertenecen al freno, mientras que los del rango de [0;1] conciernen al acelerador (Figura 10). Los rangos definidos, tanto para la salida, como para las entradas fueron estable-

cidos de manera experimental, observando cómo se comportaba el sistema a medida que se modificaban los valores.

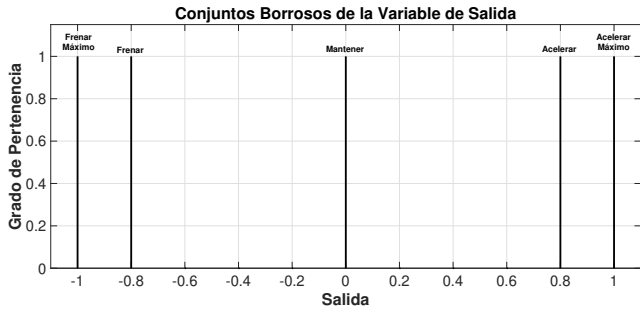


Figura 10: Función de pertenencia de los actuadores.

En la Tabla 1 se presentan las reglas que constituyen al controlador. A las cuales se les agregaron dos reglas más, para incrementar las acciones sobre el controlador en los casos extremos, es decir, cuando se tenga que frenar de emergencia, o acelerar hasta el máximo para alcanzar la referencia:

IF *Error_Vel* Muy Negativo THEN *acc/brk* Frenar Máximo

IF *Error_Dist* Muy Positivo THEN *acc/brk* Acelerar Máximo

Tabla 1: Reglas del controlador borroso.

Error de Velocidad	Error de Distancia		
	Negativo	Central	Positivo
Negativo	Frenar Máximo	Frenar	Mantener
Central	Frenar Máximo	Mantener	Acelerar
Positivo	Mantener	Acelerar	Acelerar Máximo

A partir de la configuración completa de las salidas, junto a las reglas y a las funciones de pertenencia se obtiene la superficie de control mostrada en la Figura 11. A través de dicha superficie se puede predecir el comportamiento del controlador, además de observar que no se presenta ningún tipo de discontinuidad que pueda poner en riesgo a las maniobras.

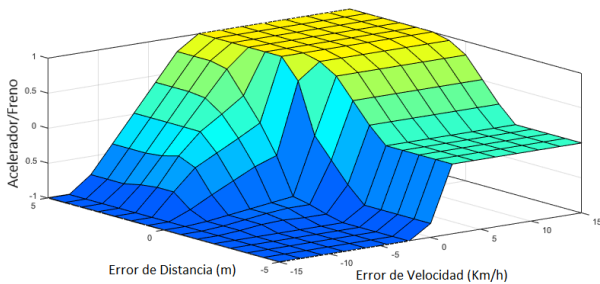


Figura 11: Superficie de control.

3.2. Control Lateral

Para el seguimiento de trayectoria, se utiliza un controlador lateral cuyo objetivo es seguir la referencia de ruta recorrida por

el vehículo líder. Al igual que en las maniobras anteriores, para el diseño del controlador se hizo uso del Toolbox de MATLAB para controladores borrosos.

3.2.1. Algoritmo de control

Como entrada del sistema se emplearon dos variables: 1) el *error lateral*, y 2) el *error angular* con respecto a la trayectoria, mientras que para la salida se estableció una, la posición del volante en grados (°). Para la obtención de los errores se emplea como referencia la posición del vehículo líder, con la cual se construye un vector de puntos, que es actualizado a una frecuencia preestablecida de 50 ms (Figura 12). Con dicho vector se busca el segmento de recta que se encuentre más cerca del coche, el cual es comparado con la posición proyectada (x_{vs}, y_{vs}) hasta las ruedas delanteras del vehículo (*lookahead*) (Ecuación 5).

$$\begin{aligned} x_{vs} &= lookahead \cdot \cos(heading) \\ y_{vs} &= lookahead \cdot \sin(heading) \end{aligned} \quad (5)$$

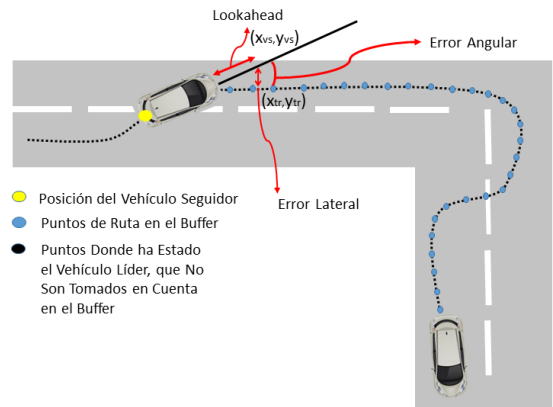


Figura 12: Características del Control Lateral implementado

El *error lateral* (e_{lat}) se define como la distancia entre la posición del vehículo (x_{vs}, y_{vs}), y el punto más cercano a la trayectoria (calculado a partir de la proyección ortogonal sobre la ruta, (x_{tr}, y_{tr})) (Ecuación 6). El signo del error se calcula con el producto escalar de los vectores formados por la posición del vehículo y del segmento de recta. Si el signo es positivo, indica que el coche se encuentra a la derecha de la recta.

$$e_{lat} = \sqrt{(x_{vs} - x_{tr})^2 + (y_{vs} - y_{tr})^2} \quad (6)$$

Con respecto al cálculo del *error angular* (e_{ang}), se debe tomar en consideración, que el mismo es la desviación del ángulo del vehículo (ψ_{vs}) con respecto a la trayectoria (ψ_{tr}), la cual puede tomar valores entre $[-\pi; \pi]$. Así pues, el error se calcula a través de la siguiente fórmula:

$$e_{ang} = \arctan \frac{\sin \psi_{vs}}{\cos \psi_{vs}} - \arctan \frac{\sin \psi_{tr}}{\cos \psi_{tr}} \quad (7)$$

3.2.2. Diseño del controlador borroso

Para el *error lateral* se establecieron tres funciones de pertenencias (Figura 13), con una apertura central amplia, de rango $[-0.6; 0.6]$ metros.

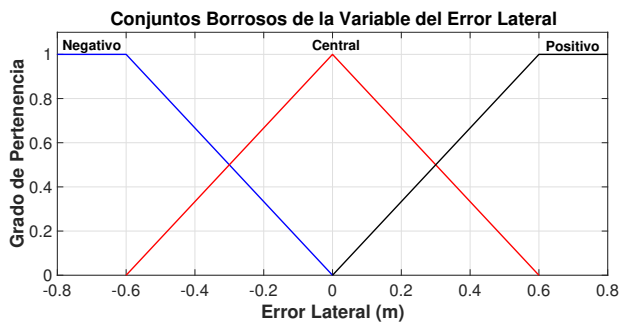


Figura 13: Función de pertenencia del error lateral.

En el caso del *error angular* se definieron, al igual que en el error lateral tres funciones de pertenencia con una apertura central amplia (Figura 14), con la diferencia que el rango es de [-20;20] grados.

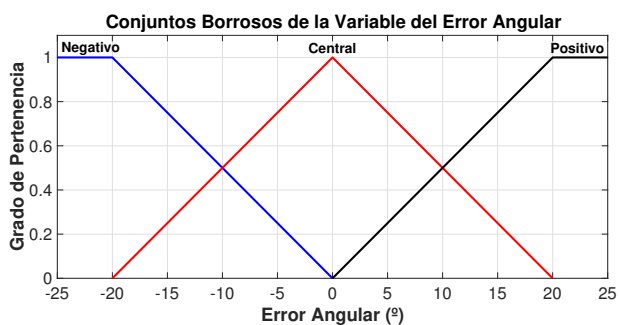


Figura 14: Función de pertenencia del error angular.

Como se mencionó anteriormente, la salida del controlador corresponde a la señal normalizada sobre el volante, para la cual se definieron 9 valores del tipo *singleton* (Figura 15), dentro del conjunto [-0.6;0.6]. De dichos valores, los negativos corresponden a girar el volante a la izquierda, mientras que si son positivos a girar el volante a la derecha. En la Tabla 2 se presentan las reglas que constituyen al controlador.

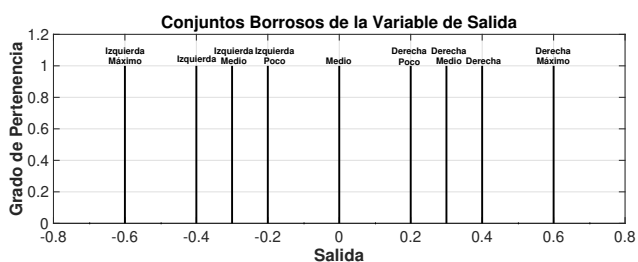


Figura 15: Función de pertenencia del actuador.

Tabla 2: Reglas del controlador difuso.

	Error Lateral		
Error Angular	Negativo	Central	Positivo
Negativo	Izquierda Máximo	Izquierda	Izquierda Medio
Centra	Izquierda Poco	Central	Derecha Poco
Positivo	Derecha Medio	Derecha	Derecha Máximo

Adicionalmente, a partir de la configuración de las salidas, junto a las reglas y a las funciones de pertenencia se obtiene la superficie de control mostrada en la Figura 16, a través de la cual puede predecir el comportamiento del controlador.

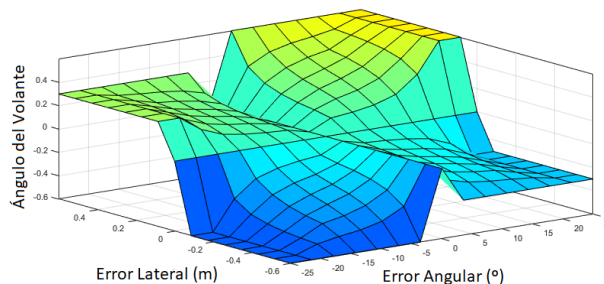


Figura 16: Superficie de control.

4. Resultados

Las pruebas experimentales se llevaron a cabo en las instalaciones de TecNALIA Research & Innovation (Figura 17), que cuenta con un circuito cerrado de 80 m de largo y 15 m de ancho, compuesto por un tramo recto, dos curvas y un tramo con cambio de carril. El escenario se implementó en simulación con las mismas características. Para validar las tres maniobras, se implementó cada una en un entorno de prueba diferente, como los presentados en el marco cooperativo (virtual, mixto y real) con el fin de observar el desempeño de los controladores en distintas configuraciones del entorno.

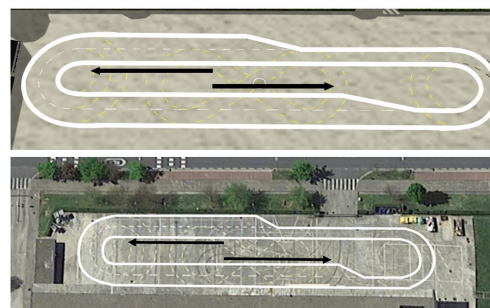
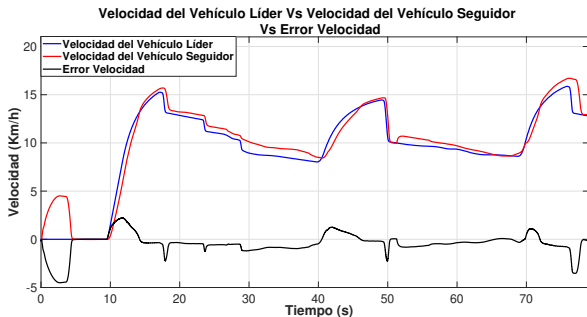


Figura 17: Pista virtual y real de TecNALIA Research & Innovation.

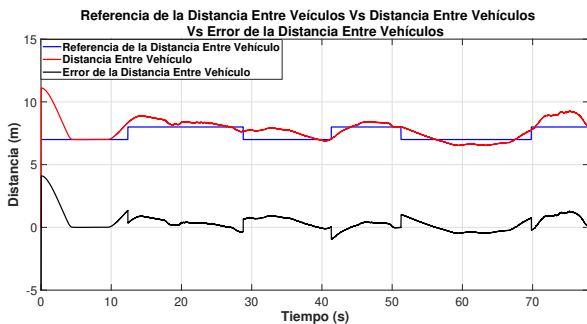
4.1. Entorno Virtual

La prueba realizada en este entorno busca validar la maniobra ACC junto con el Control Lateral en escenarios urbanos, para lo cual se emplean dos vehículos virtuales en ordenadores distintos, donde uno ejerce el rol de líder, recorriendo la pista variando su velocidad hasta un máximo de 15 km/h, mientras que el segundo cumple la función del seguidor, el cual es el encargado de efectuar la maniobra. Cabe destacar, que el vehículo seguidor comienza la prueba detrás de la distancia de referencia, para que al iniciar la maniobra se acerque a esta y luego avance en el momento en el que el otro coche lo haga. Primeramente se presenta el comportamiento del controlador del ACC, el cual puede ser observado en las Figuras 18a y 18b. La primera corresponde a la comparación entre las velocidades de ambos vehículos, donde se puede apreciar que el coche seguidor sigue en todo momento a la velocidad del vehículo líder. Además, al

inicio acelera hasta alcanzar la distancia de referencia, para luego frenar y esperar a que el otro coche avance. En relación con la separación entre los vehículos se puede apreciar en la Figura 18b como el coche sigue a la referencia planteada con un error máximo de 20 cm al estabilizarse, lo que equivale al 2% de error.



(a) Gráfica de la velocidad de los vehículos.



(b) Gráfica de la distancia entre vehículos.

Figura 18: Comportamiento longitudinal de la maniobra de ACC con seguidor de trayectoria

Con respecto al desempeño del controlador del Control Lateral, se puede observar en la Figura 19 que la ruta seguida por los vehículos es similar, diferenciándose en menor medida al momento de finalizar las curvas. Además, se encuentra una discrepancia con respecto a la ruta generada por el planificador local, ya que los ajustes realizados son en base a una posición que ya trae consigo un error, que es la obtenida a través del vehículo líder.

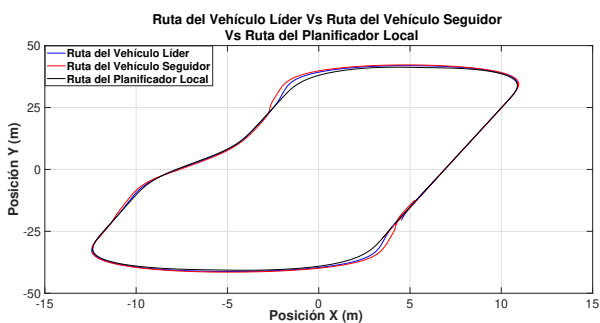
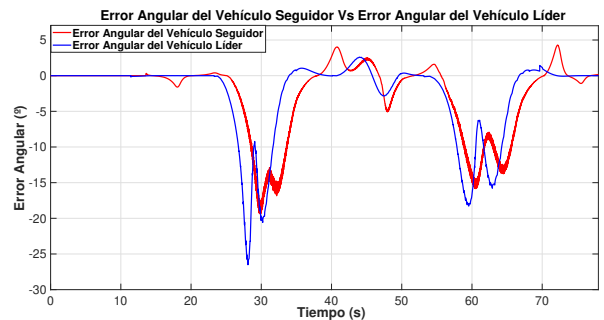


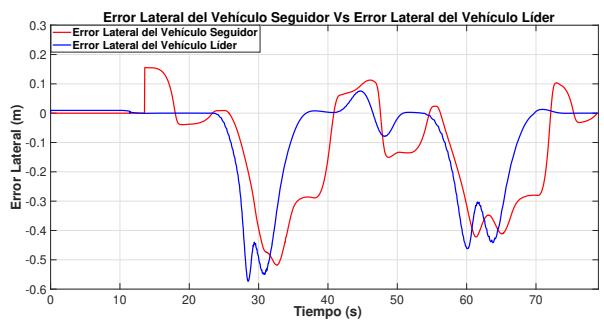
Figura 19: Gráfica de la ruta seguida por los vehículos.

Las Figuras 20a y 20b, muestran el comportamiento de los errores angular y lateral de los vehículos. En dichas gráficas se observa que los errores máximos presentados son en las curvas,

los cuales poseen un valor de 18° y 52 cm, para la primera curva y 17° y 42 cm para la segunda. Adicionalmente, se puede contemplar cómo los errores del vehículo seguidor son menores a los presentados por el líder, siendo la máxima diferencia para el error lateral, de 10 cm y, 20° para el error angular. Si bien, no son comparados bajo la misma referencia, demuestran cómo el coche logra un recorrido bastante similar.



(a) Gráfica del error lateral de los vehículos.



(b) Gráfica del error angular de los vehículos.

Figura 20: Comportamiento lateral de la maniobra de ACC con seguidor de trayectoria

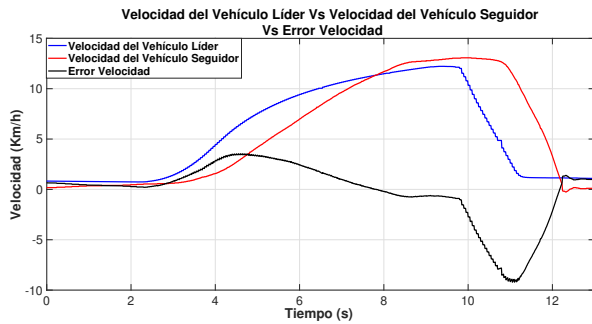
La respuesta de los controladores resultaron satisfactorias, ya que el vehículo seguidor logró mantener tanto la referencia de velocidad, como la de distancia, demostrando errores de bajo porcentaje, así como también pudo realizar un recorrido similar al del vehículo líder, con errores de baja magnitud.

4.2. Entorno Mixto

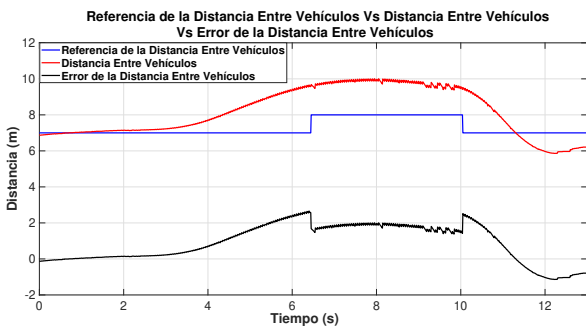
La prueba ejecutada en este entorno busca validar la maniobra de Stop & Go, para la cual se emplea un vehículo virtual como el líder, el cual una vez alcanzado una velocidad máxima de 11 Km/h, frena de forma abrupta, produciendo que el vehículo real, que es el encargado de realizar la maniobra, tenga que frenar guardando cierta distancia de seguridad. En este caso el coche seguidor comienza sobre la distancia de referencia.

En la Figura 21a se puede observar cómo el vehículo real, en un intento por alcanzar la distancia de referencia, excede la velocidad establecida hasta alcanzar unos 13 Km/h. A continuación, frena en pro de mantener la distancia entre los coches. Con respecto a la distancia, se puede apreciar en la Figura 21b, como el vehículo busca alcanzar la referencia. En el momento de ejecutar el frenado, él mismo finaliza a 6 m de distancia con respecto al líder, por lo que excede la distancia de seguridad en

1 m. Si bien, se tiene un error del 14 %, la separación que guarda en relación al coche virtual es considerada aceptable, ya que no se corre ningún riesgo de accidente.



(a) Gráfica de la velocidad de los vehículos.



(b) Gráfica de la distancia entre vehículos.

Figura 21: Comportamiento longitudinal de la maniobra de Stop & Go

4.3. Entorno Real

La prueba realizada en este entorno busca validar la maniobra de ACC sin seguidor de ruta, para lo cual se emplean dos vehículos reales, uno instrumentado con sensores y sistema de comunicación conducido por una persona, que ejerce la función de líder y uno completamente automatizado que cumple el rol del seguidor, el cual es el encargado de ejecutar la maniobra. En este caso la velocidad máxima alcanzada fue de unos 22 Km/h y de igual forma el vehículo seguidor comienza por detrás de la distancia de referencia.

El comportamiento con respecto a la velocidad del vehículo puede ser encontrado en la Figura 22, donde se aprecia que el coche sigue de forma correcta a la velocidad del líder. Además, realizando los ajustes necesarios para poder alcanzar la referencia de la distancia.

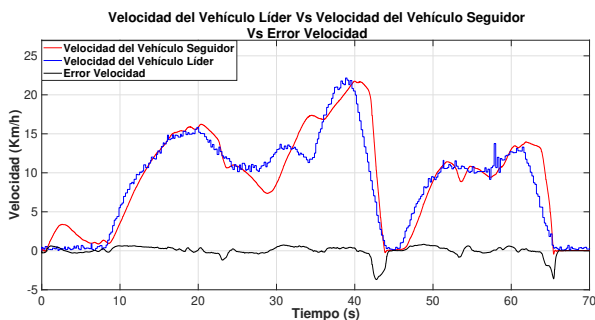


Figura 22: Gráfica de la velocidad de los vehículos.

En lo que respecta a la distancia en la Figura 23, se aprecia cómo se busca mantener la referencia lo más cercana posible. Sin embargo, debido a la manera de calcular la distancia entre los vehículos, que es suponiendo una línea recta, se encuentran errores de mayor magnitud al momento de tomar las curvas, lo cual sumado al hecho de que se pueden producir cambios bruscos de velocidad en dichos segmentos, y a los posibles retrasos en la comunicación en el ámbito real, el error se puede ver aumentado aún más, como es el caso de la primera curva, el cual alcanza un valor de 6 m, representando un error del 75 %.

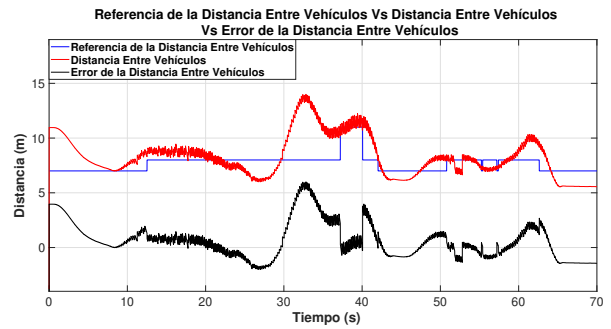


Figura 23: Gráfica de la distancia entre vehículos.

Si bien se presenta un error de mayor magnitud con respecto a la distancia en un tramo, el controlador muestra una respuesta satisfactoria en el resto del recorrido, siguiendo en buena medida a la referencia de velocidad, así como también a la de distancia, demostrando además, seguridad al ejecutar la maniobra ya que, a pesar del error, nunca presentó riesgo de accidente.

5. Conclusiones

En este trabajo se presenta un marco para maniobras cooperativas en vehículos automatizados. La validación de este enfoque se realizó con tres escenarios distintos y tres maniobras. Más específicamente las maniobras son: ACC, Stop & Go y Control Lateral. Dichos entornos son: real, virtual y mixto, siendo este último la unión de un vehículo virtual que cumple la función de líder y uno real que efectúa la del seguidor, basado en un controlador borrosos. Los ensayos planteados para la validación de las maniobras en los distintos entornos mostraron resultados satisfactorios, demostrando de esta forma la eficacia del marco cooperativo presentado para las pruebas de las mismas, ya sean en escenarios reales, virtuales o mixtos.

Los experimentos en el entorno virtual fueron los que mejor resultado mostraron, el vehículo pudo realizar el seguimiento completo del coche líder, manteniendo la distancia de seguridad así como la velocidad dictada por el mismo, de la misma forma ocurre con la ruta fijada por el vehículo líder, la cual pudo seguir correctamente. Este comportamiento es de esperar ya que en dicho entorno no son contemplado varias variables existentes en el ámbito real, por lo que la maniobra se puede ejecutar de mejor forma.

Sin embargo, a pesar de que en el entorno real las maniobras se pueden ver afectadas por distintos factores, como lo pueden ser el retraso en las comunicaciones con el otro vehículo, o en los actuadores, el coche pudo ejecutar la maniobra eficazmente guardando la distancia de seguridad establecida manteniendo la

velocidad fijada por el vehículo líder. De la misma forma ocurre en el entorno mixto, si bien se vió afectado por dichas variables el coche real pudo efectuar el frenado con cierta distancia de separación con respecto al vehículo virtual.

De esta forma, los resultados obtenidos permiten la posibilidad para el desarrollo y validación de distintas maniobras, empleando las metodologías descritas en el presente trabajo. Como trabajos futuros es de interés realizar una caracterización de los actuadores del vehículo real e incluirlos en el modelo del vehículo simulado, donde se tomen en consideración los tiempos de retrasos en la ejecución de los comandos. Adicionalmente, este marco podría ser utilizado para la validación de otras maniobras, como lo son: intersecciones, adelantamientos con más de un vehículo, toma de decisiones en rotondas, entre otras, así como la optimización de los controladores de las maniobras mostradas. Por otra parte, es importante investigar e implementar nuevas tecnologías para la comunicación entre vehículos, que permitan una conexión más robusta y de esta forma reducir los errores que se puedan ocasionar por pérdida de datos, desconexiones, retrasos, entre otros.

Agradecimientos

Los autores quieren agradecer al proyecto SerIoT H2020 (Con número de concesión 780139) por su apoyo en el desarrollo de este trabajo.

Referencias

- Cuadrado, J., Vilela, D., Iglesias, I., Martín, A., Peña, A., 2013. A multibody model to assess the effect of automotive motor in-wheel configuration on vehicle stability and comfort. *ECCOMAS Multibody Dynamics*.
- Da Cunha, F. D., Boukerche, A., Villas, L., Viana, A. C., Loureiro, A. A., 2014. Data communication in vanets: a survey, challenges and applications. Ph.D. thesis, INRIA Saclay; INRIA.
DOI: 10.1016/j.adhoc.2016.02.017
- Driankov, D., Hellendoorn, H., Reinfrank, M., 1996. Introduction. In: *An Introduction to Fuzzy Control*. Springer, pp. 1–36.
DOI: 10.1007/978-3-662-03284-8
- During, M., Lemmer, K., 2016. Cooperative maneuver planning for cooperative driving. *IEEE Intelligent Transportation Systems Magazine* 8 (3), 8–22.
DOI: 10.1109/ITS.2016.2549997
- Española, D. G. D. T., Jul. 2015. Seguridad vial. <http://www.dgt.es/es/seguridad-vial/>.
- Fernandez, S., Ito, T., nov 2016. Driver classification for intelligent transportation systems using fuzzy logic. In: *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE.
DOI: 10.1109/itsc.2016.7795711
- Festag, A., 2014. Cooperative intelligent transport systems standards in europe. *IEEE communications magazine* 52 (12), 166–172.
DOI: 10.1109/MCOM.2014.6979970
- González, D., Pérez, J., Milanés, V., Nashashibi, F., 2016. A review of motion planning techniques for automated vehicles. *IEEE Transactions on Intelligent Transportation Systems* 17 (4), 1135–1145.
DOI: 10.1109/TITS.2015.2498841
- Iancu, I., 2012. A mamdani type fuzzy logic controller. In: *Fuzzy Logic-Controls, Concepts, Theories and Applications*. InTech.
DOI: 10.5772/36321
- Jena, K. S., Joseph, A. V., Senapati, P. R. R., 2016. Fuzzy logic based approach for controlling of a vehicle in its longitudinal motion. *Middle-East Journal of Scientific Research* 24 (S1), 346–352.
DOI: 10.5829/idosi.mejsr.2016.24.S1.62
- Lattarulo, R., González, L., Martí, E., Matute, J., Marcano, M., Pérez, J., 2018. Urban motion planning framework based on n-bézier curves considering comfort and safety. *Journal of Advanced Transportation* 2018.
DOI: 10.1155/2018/6060924
- Lattarulo, R., Marcano, M., Pérez, J., 2017a. Overtaking maneuver for automated driving using virtual environments. In: *International Conference on Computer Aided Systems Theory*. Springer, pp. 446–453.
DOI: 10.1007/978-3-319-74727-9_54
- Lattarulo, R., Pérez, J., Dendaluce, M., 2017b. A complete framework for developing and testing automated driving controllers. *IFAC-PapersOnLine* 50 (1), 258–263.
DOI: 10.1016/j.ifacol.2017.08.043
- Lin, W.-Y., Li, M.-W., Lan, K.-C., Hsu, C.-H., 2010. A comparison of 802.11 a and 802.11 p for v-to-i communication: A measurement study. In: *International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness*. Springer, pp. 559–570.
DOI: 10.1007/978-3-642-29222-4_39
- 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. *Complexity* 2018.
DOI: 10.1155/2018/7615123
- Marzbanrad, J., Moghaddam, T.-z., et al., 2015. Prediction of drivers accelerating behavior in the stop and go maneuvers using genetic algorithm-artificial neural network hybrid intelligence. *International Journal of Automotive Engineering* 5 (2), 986–998.
- Milanés, V., Villagrà, J., Godoy, J., González, C., 2012. Comparing fuzzy and intelligent pi controllers in stop-and-go manoeuvres. *IEEE Transactions on Control Systems Technology* 20 (3), 770–778.
DOI: 10.1109/TCST.2011.2135859
- Milanés, V., Villagrà, J., Godoy, J., Simo, J., Pérez, J., Onieva, E., 2012. An intelligent v2i-based traffic management system. *IEEE Transactions on Intelligent Transportation Systems* 13 (1), 49–58.
DOI: 10.1109/TITS.2011.2178839
- Morcillo, C. G., 2011. Lógica difusa. una introducción práctica. Universidad de Castilla-La Mancha.
- Pérez, J., Milanés, V., Alonso, J., Onieva, E., De Pedro, T., 2010. Adelantamiento con vehículos autónomos en carreteras de doble sentido. *Revista Iberoamericana de Automática e Informática Industrial RIAI* 7 (3), 25–33.
DOI: 10.1016/S1697-7912(10)70039-X
- Rastelli, J. P., Peñas, M. S., 2015. Fuzzy logic steering control of autonomous vehicles inside roundabouts. *Applied Soft Computing* 35, 662–669.
DOI: 10.1016/j.asoc.2015.06.030
- Rizvi, R., Kalra, S., Gosalia, C., Rahnamayan, S., 2014. Fuzzy adaptive cruise control system with speed sign detection capability. In: *Fuzzy Systems (FUZZ-IEEE), 2014 IEEE International Conference on*. IEEE, pp. 968–976.
DOI: 10.1109/FUZZ-IEEE.2014.6891748
- Schmied, R., Moser, D., Waschl, H., del Re, L., 2016. Scenario model predictive control for robust adaptive cruise control in multi-vehicle traffic situations. In: *Intelligent Vehicles Symposium (IV), 2016 IEEE*. IEEE, pp. 802–807.
DOI: 10.1109/IVS.2016.7535479
- Sriranjan, S., Lattarulo, R., Pérez-Rastelli, J., Ibanez-Guzman, J., Pena, A., 2017. Lateral controllers using neuro-fuzzy systems for automated vehicles: A comparative study.
- Sugeno, M., 1985. An introductory survey of fuzzy control. *Information sciences* 36 (1-2), 59–83.
DOI: 10.1016/0020-0255(85)90026-X
- Sugeno, M., Murakami, K., 1984. Fuzzy parking control of model car. In: *Decision and Control, 1984. The 23rd IEEE Conference on*. Vol. 23. IEEE, pp. 902–903.
DOI: 10.1109/CDC.1984.272144
- Villagrà, J., Milanés, V., Pérez, J., González, C., 2010. Model-free control techniques for stop & go systems. In: *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on*. IEEE, pp. 1899–1904.
DOI: 10.1109/ITSC.2010.5625228
- Wang, X., Fu, M., Ma, H., Yang, Y., 2015. Lateral control of autonomous vehicles based on fuzzy logic. *Control Engineering Practice* 34, 1–17.
DOI: 10.1016/j.conengprac.2014.09.015
- Zohdy, I. H., Kamalanathsharma, R. K., Rakha, H., 2012. Intersection management for autonomous vehicles using icacc. In: *Intelligent Transportation Systems (ITSC), 2012 15th International IEEE Conference on*. IEEE, pp. 1109–1114.
DOI: 10.1109/ITSC.2012.6338827