



**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**  
**FACULTAD DE INGENIERÍA**

**Diseño del Control de Crucero  
Adaptativo de un Vehículo Terrestre  
Autónomo Mediante un Ambiente  
Simulado**

**TESIS**

Que para obtener el título de  
**Ingeniero Mecatrónico**

**P R E S E N T A**

Ricardo Alberto Miranda Martínez

**DIRECTOR DE TESIS**

Dr. Octavio Díaz Hernández



Ciudad Universitaria, Cd. Mx., 2022



# *Resumen*

El presente proyecto de investigación desarrolla cuatro estrategias de control para el diseño de un control de crucero adaptativo, entre las que se encuentran: control proporcional-integral-derivativo, control de modo deslizante, control por constante de tiempo-espacio y el control de modelo predictivo.

El modelo del control de crucero adaptativo (ACC) propuesto se compone de un modelo no lineal de la dinámica vehicular longitudinal y lateral; así como, un modelo trivial de la dinámica del tren motriz. Dicho modelo es implementado mediante el uso de una plataforma de simulación, *Matlab y Simulink*, con el propósito de llevar a cabo una representación visual de la estrategia de control en conjunto con un motor gráfico, en este caso *Unreal Engine*. Así mismo, el modelo utilizado se descompone en los sistemas y subsistemas esenciales de un vehículo autónomo, en el caso del ACC se analiza con mayor detalle la parte superior del controlador cuya función es generar la aceleración deseada para que el controlador inferior pueda realizar el seguimiento de la misma en una situación de frenado o aceleración.

El enfoque principal del estudio es la comparación de cada estrategia de control mediante el uso de seis escenarios estandarizados mediante la norma normatividad internacional ISO 15622 e ISO 22178. El propósito de hacer uso de estos escenarios es comprender su funcionamiento y determinar cual de ellas presenta una mejor respuesta ante diversas situaciones. Basándose en los resultados obtenidos y el análisis desarrollado, se concluye que para el modelo propuesto cada estrategia cumple con el objetivo de control al evitar un choque entre los vehículos propuestos y mantener una velocidad objetivo durante el recorrido. Además, cada estrategia presenta características y comportamientos particulares ante los cambios en los parámetros de diseño; sin embargo, en general el controlador MPC destaca sobre el resto de estrategias por su desempeño en la mayoría de los escenarios.



# *Agradecimientos*

A mis Padres,

Por su incondicional apoyo en cada proyecto, por estar siempre a mi lado en los momentos difíciles durante mi vida, por compartir su espíritu de perseverancia, por ser mi inspiración día con día. Hoy que concluyo mis estudios, dedico este logro a ustedes.

Gracias por su amor incondicional.

A mi hermano,

Por ser un ejemplo a seguir, por cada minuto de tiempo invertido en mi desarrollo para continuar mi camino.

Gracias por tu comprensión y consejos.

A la Universidad Nacional Autónoma de México,

Gracias por permitirme formar parte de tu comunidad, por ser mi segundo hogar, por todas las experiencias aprendidas, por las oportunidades que me has otorgado y por permitirme desarrollar mi pasión por la ingeniería.

Al Dr. Octavio Díaz Hernández,

Gracias por todos sus consejos, constancia e interés en mi desarrollo profesional y personal.

A mis compañeros y amigos,

Por ser parte de una historia de cinco años, por hacer de esta experiencia algo único e irrepetible.

Gracias por estar siempre allí.



# Contenido

<i>1. Introducción</i>	2
1.1 Una idea posible	2
1.2 Objetivos	3
1.3 Estructura del escrito	3
<i>2. Vehículos autónomos</i>	6
2.1 Antecedentes	6
2.2 ¿Qué es un vehículo autónomo?	7
2.3 ¿Cuál es la importancia de los vehículos autónomos?	8
2.4 Sistemas Avanzados de Asistencia al Conductor (ADAS)	8
2.4.1 Sistemas de control lateral	9
2.4.2 Sistemas de control longitudinal	11
2.4.3 Asistentes de estacionado	13
2.4.4 Sistemas de visión aumentada	14
2.4.5 Sistemas de adaptación de velocidad inteligentes	15
2.5 Niveles de automatización	15
2.6 Sensores	17
2.6.1 Radar	18
2.6.2 Sensor Ultrasónico	19
2.6.3 LiDAR	20
2.6.4 Cámaras	21
2.6.5 GNSS	22
2.6.6 IMU	23
2.6.7 Odómetros	24
2.7 Ejemplos de vehículos autónomos	25
2.7.1 Waymo	25
2.7.2 GM Cruise	27

2.7.3 Argo AI	29
2.8 Conclusión	30
<i>3. Simuladores</i>	33
3.1 ¿Qué es un simulador?	33
3.2 Campos de aplicación de los simuladores	33
3.3 Métodos de simulación	34
3.3.1 Model in the Loop-MIL	35
3.3.2 Software in the Loop-MIL	36
3.3.3 Hardware in the Loop-HIL	37
3.3.4 Vehicle in the Loop-VIL	37
3.4 Plataformas de simulación	38
3.4.1 PaTAVTT	38
3.4.2 Mathworks, Matlab & Simulink	39
3.4.3 National Instrument, Monodrive	40
3.4.4 CarSim	41
3.4.5 Carla	43
3.4.6 SVL	44
3.5 Selección de plataforma de simulación	45
3.6 Conclusión	47
<i>4. Diseño del Control de Crucero Adaptativo (ACC)</i>	49
4.1 Definición del problema	49
4.2 Propuesta de Solución	50
4.3 Conclusión	50
<i>5. Modelado del Sistema</i>	52
5.1 Dinámica vehicular longitudinal	52
5.1.2 Fuerza aerodinámica	53
5.1.3 Resistencia a la Rodadura	53
5.1.4 Fuerza longitudinal del neumático	54
5.2 Modelo dinámico de dos ruedas (Bicycle Model)	57
5.3 Control de Crucero Adaptativo	64
5.3.1 Arquitectura del controlador	64
5.4 Estrategias de control	65



5.4.1 Control Clásico (PID)	66
5.4.2 Constant Time-Gap Law	68
5.4.3 Sliding Mode Control	69
5.4.4 MPC	71
5.5 Conclusión	76
<b>6. Banco de pruebas Virtual</b>	<b>79</b>
6.1 Descripción del banco de pruebas	79
6.1.1 Simulation 3D Scenario	80
6.1.2 Lane Marker Detector	81
6.1.3 Vehicle Detector	82
6.1.4 Forward Vehicle Sensor Fusion	82
6.1.5 Lane Following Decision Logic	83
6.1.6 Lane Following Controller	83
6.1.7 Vehicle Dynamics	84
6.1.8 Metrics Assesment	85
6.2 Conclusión	86
<b>7. Metodología</b>	<b>88</b>
7.1 Normatividad Internacional ISO 15622 e ISO 22178	88
7.2 Escenarios/Casos de prueba	89
7.3 Conclusión	94
<b>8. Resultados</b>	<b>96</b>
8.1 Selección de Parámetros	96
8.1.1 Clásico PID	96
8.1.2 Constant-Time Gap Law (CTG)	99
8.1.3 Sliding mode controller	104
8.1.4 MPC Model Predictive Control	109
8.2 Análisis de Resultados	122
8.2.1 Escenario 1	123
8.2.2 Escenario 2	125
8.2.3 Escenario 3	138
8.2.4 Escenario 4	153
8.2.5 Escenario 5	166

8.2.6 Escenario 6	179
<i>9. Conclusión</i>	193
<i>10. Referencias</i>	1



# 1. *Introducción*

## 1.1 *Una idea posible*

Muchos advenimientos tecnológicos surgieron y fueron objeto de escepticismo, algunos de ellos metas lejanas en cada uno de sus contextos históricos; sin embargo, en la actualidad podemos observar la manera en la que el ingenio humano ha sido capaz de innovar y superar esos obstáculos para desarrollar nuevas tecnologías.

Es muy común escuchar la idea de un auto que se conduce por sí mismo, esta idea surgió mucho tiempo atrás, fue demeritada y considerada como algo imposible; no obstante, al pasar de los años el desarrollo e implementación de nuevas tecnologías ha hecho posible el funcionamiento de los llamados vehículos autónomos. Su implementación ha sido paulatina y continúa desarrollándose día con día, siendo impulsada principalmente por las industrias tecnológicas interesadas en la puesta en operación de dichos vehículos, tal y como es la industria automovilística.

Hoy en día, la idea de hacer uso de vehículos autónomos resulta bastante atractiva, ya sea como uso de transporte de carga, de personas o como vehículo privado, con el objetivo de llegar de un lugar a otro con la seguridad, comodidad y confianza que estos pueden llegar a ofrecer. Basta pensar en otras posibles características como el aprovechamiento del tiempo de traslado, la reducción de accidentes automovilísticos e inclusive la posible disminución de emisiones contaminantes en nuestro planeta.

Con el objetivo de llevar a cabo el desarrollo de estos vehículos y aprovechar dichas características existen ciertos factores a considerar. Tal y como los seres humanos obtenemos información de nuestro entorno, los vehículos autónomos necesitan replicar este comportamiento, es por ello que actualmente la gran mayoría de vehículos comerciales son implementados con sistemas dedicados a reconocer, analizar y ejecutar acciones de manera automática con respecto a su entorno, tales como el control de cruceo adaptativo considerado como parte de los sistemas avanzados de asistencia a la conducción (ADAS). Por esta misma razón se requiere entender, analizar y mejorar el desempeño de dichos sistemas y, consecuentemente, el desempeño general de los vehículos autónomos.

Lo anterior conlleva realizar exhaustivas pruebas de funcionamiento de cada uno de los sistemas y subsistemas que componen a los vehículos autónomos aumentando el costo de su desarrollo; sin embargo, una de las posibles soluciones es mediante el uso de simuladores. Estos son de gran importancia dado que facilitan enormemente el desarrollo del software requerido en el desarrollo de estos vehículos, además favorecen la simulación de diferentes escenarios de prueba, en los cuales se pueden mejorar aspectos importantes de la tarea de conducción de un vehículo, tales como la seguridad, velocidad de respuesta y confiabilidad.

## 1.2 Objetivos

El presente escrito centra su atención en cumplir con los siguientes objetivos:

- Analizar e identificar los diferentes tipos de controladores utilizado en el desarrollo de los sistemas de asistencia en la conducción de los vehículos, en específico en el desarrollo del controlador de cruceo adaptativo.
- Presentar una herramienta útil para simularlos y estudiarlos.
- Elaborar e implementar herramientas de diseño del controlador de cruceo adaptativo en un simulador 3d para pruebas de los controladores.
- Realizar pruebas de simulación de los modelos propuestos en cada sistema para los seis escenarios de los que se compone el presente estudio.
- Determinar y proponer qué tipo de controlador es el más adecuado para un control de cruceo adaptativo.
- Evaluar el desempeño funcional de cada metodología de control.
- Fundar las bases para un proyecto de investigación con futura expansión.

## 1.3 Estructura del escrito

El presente trabajo de investigación está compuesto por 10 capítulos en los cuales se aborda la problemática planteada anteriormente, tal y como se enlista a continuación:

- **Capítulo 1:** En este capítulo se presentan los objetivos generales de la investigación, así como la estructura del escrito y el contenido de cada capítulo.
- **Capítulo 2:** En esta sección se expone el estado del arte de los vehículos autónomos, los principales componentes y sistemas que lo conforman; así como el avance actual en su implementación comercial.
- **Capítulo 3:** En este apartado se revisa el estado actual de las plataformas de simulación/simuladores utilizados para el desarrollo y diseño de los sistemas que conforman los vehículos autónomos.
- **Capítulo 4:** En este título se define la problemática sobre la cual se fundamenta el diseño de los controladores, así como la descripción de la solución detallada en los capítulos subsecuentes.

- **Capítulo 5:** En este capítulo se proponen diferentes modelos para realizar el diseño de un sistema de control de cruce adaptativo, mediante diferentes metodologías de control y el uso de la plataforma de simulación seleccionada en el Capítulo 3.
- **Capítulo 6:** En esta sección se describe en detalle los sistemas y subsistemas que componen la plataforma de simulación/ banco de pruebas virtual determinado en el capítulo tres y utilizado para realizar las simulaciones correspondientes de los modelos propuestos en el Capítulo 5.
- **Capítulo 7:** En este apartado se describe la metodología utilizada para llevar a cabo la evaluación del desempeño, así como la comparación de los modelos y controladores propuestos en el Capítulo 5.
- **Capítulo 8:** En este capítulo se detalla el proceso empleado para determinar los parámetros de los controladores propuestos, así como los resultados de los diferentes escenarios de prueba utilizados para determinar el mejor comportamiento.
- **Capítulo 9:** En esta sección se presentan las conclusiones generales del presente escrito.
- **Capítulo 10:** Referencias.



## 2. Vehículos autónomos

### 2.1 Antecedentes

A lo largo de la historia han existido diversos avances tecnológicos que han sido desarrollados por los humanos, tal es el caso del automóvil, este ha sufrido una serie de cambios desde su invención, esto se puede observar si consideramos a la máquina de vapor y el motor a gas como sus más remotos antepasados, los cuales fueron desarrollados a mediados del siglo XVII, esto desencadenó años después la invención de los primeros carruajes de vapor; posteriormente, se llevó a cabo la invención del motor moderno de gasolina en 1860 y fue implementado de manera comercial en 1878; más adelante, el automóvil continuo evolucionando a tal punto que se implementaron vehículos totalmente eléctricos en el 2003 por la referente compañía americana Tesla [1][2]. Por su parte el control sufrió un cambio importante con el nacimiento de la teoría moderna del control que nace a partir del año 1955 basada en la construcción de un sistema de estudio en variables de estado y haciendo uso del dominio del tiempo. Así mismo, la posterior implementación y expansión del uso del computador en la década de los setentas significó un salto tecnológico en las industrias, teniendo aplicaciones relevantes en el uso de adquisición de datos, supervisión, control secuencial, control analógico digital, control digital directo y análisis de datos. Para el año de 1999 la mayor parte de procesos industriales eran controlados mediante un computador y hoy en día se conforman mayormente por el uso de plataformas computacionales, las cuales son fundamentales para hacer funcionar sistemas de hardware y/o software en conjunto [3][4].

Los vehículos autónomos hacen uso de la combinación e integración de una gran variedad de áreas del conocimiento, disciplinas como ciencias de la computación, ingeniería mecánica, ingeniería eléctrica, ingeniería de control, etc. son aquellas que hacen posible la construcción y desarrollo de este tipo tan particular de vehículos. Por tal razón su misma evolución está acompañada y es dependiente de los avances tecnológicos de cada época [5]. Basta con remitirse a los antecedentes específicos de los vehículos autónomos, desde sus primeros pasos donde se presentó la idea por la compañía GM (General Motors) en el año de 1939 en la exposición “*Futurama*”, donde se mostraba una posible visión general de 20 años en el futuro, en ella se presentaron vehículos manejando sin ayuda de un piloto, para hacer esto posible se hizo uso de circuitos integrados en la pista que fueron controlados mediante el uso de la radio [5][6]. Más adelante, en los años 80’s la Agencia de Proyectos de Investigación Avanzados de Defensa (DARPA por sus siglas en inglés) de los Estados Unidos desarrolló el proyecto Vehículo Terrestre Autónomo (ALV), este fue el primer vehículo que hizo uso de tecnología de visión computacional, LIDAR y control autónomo para planear su movimiento, llegando a alcanzar una velocidad de 3.1 km/h en terreno irregular, la estructura y componentes de este vehículo se pueden observar en la Figura 2.1.1 [5].



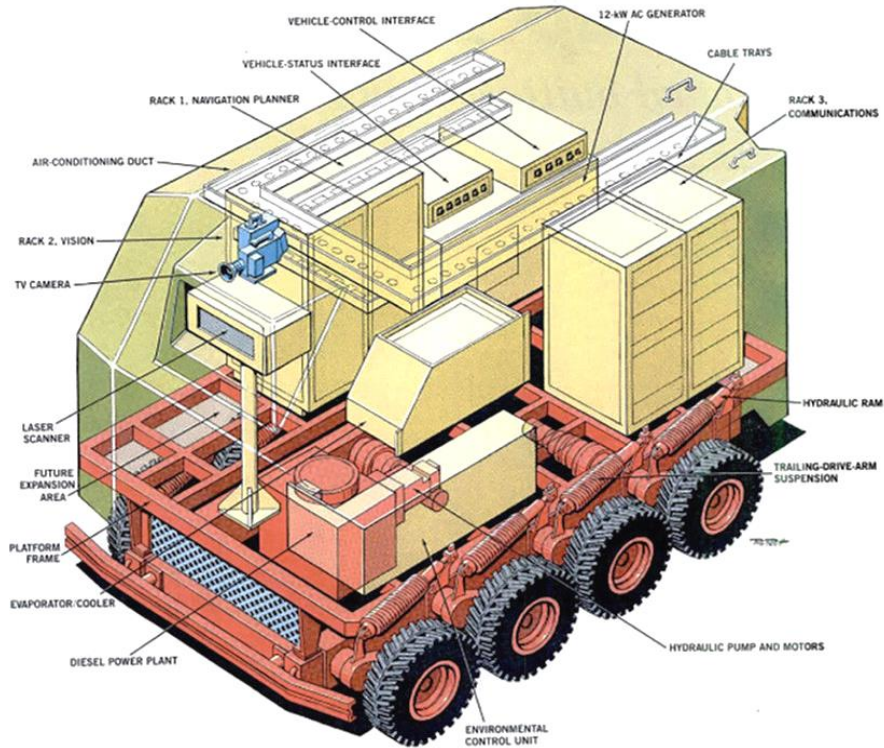


Figura 2.1.1 Ilustración de Ray Pioch. “Look, Ma! No Driver,” Popular Science 227: PP. © 1985 Ray Pioch [7]

Continuando su evolución, en el año 2005 se llevó a cabo el *DARPA Grand Challenge*, y en el 2007 el *DARPA Urban Challenge*, ambas competiciones tenían como propósito principal la construcción de vehículos autónomos que llevarían a cabo un curso específico en un tiempo determinado; adicionalmente, en el 2011 se llevó a cabo el *VisLab Intercontinental Autonomous Challenge*, que es considerada como una de las grandes competiciones de autos autónomos debido a que gracias a ella se han podido realizar avances importantes en diferentes ámbitos tecnológicos de la robótica y de los vehículos autónomos. Esta prueba realizada por los vehículos consistió de un viaje de 13 000 km el cuál se realizó con nula intervención humana, es gracias a este tipo de competiciones que se demostró la utilidad de los vehículos autónomos y que es posible su uso como medio de transporte de bienes [5][6].

## 2.2 ¿Qué es un vehículo autónomo?

De acuerdo a la definición de *Oxford Languages* se define a un **vehículo** como un aparato que se mueve a través de un medio cuya función es el transportar cosas o personas; de igual manera, **autonomía** se define como la capacidad que tiene una entidad o persona para decidir con independencia o sobre el deseo de otros; por tanto, podemos definir a un **vehículo autónomo** como un aparato que tiene la capacidad de decidir y planear sus movimientos en todo momento de manera independiente de los humanos, cuyo propósito es el de transportar personas o bienes a través de un medio específico.

Con respecto a la definición anterior es importante entender la diferencia entre un vehículo autónomo y un vehículo automatizado. La Sociedad de Ingeniero de Automoción (SAE por sus siglas en inglés) establece una diferencia entre dichos términos, esto quiere decir que *autónomo, vehículo sin conductor, auto piloto, vehículo no tripulado*, son términos que, afirman han sido empleados de manera confusa para describir sistemas de conducción automatizados o vehículos integrados con dichos sistemas. De lo anterior podemos decir que un sistema automatizado es aquel que hace uso de sistemas electrónicos o mecánicos para reemplazar la intervención humana, mientras que un sistema autónomo es aquel que aunado a la funcionalidad de un sistema automatizado es capaz de tomar decisiones de manera independiente y autosuficiente [8]. Una vez se ha establecido dicha diferencia, es importante mencionar que a pesar de que puede haber esta confusión en el uso de los términos, a lo largo de este trabajo se utilizará el término vehículo autónomo, esto debido a la gran popularidad del mismo en el ámbito cotidiano y científico.

### ***2.3 ¿Cuál es la importancia de los vehículos autónomos?***

La implementación de los vehículos autónomos, así como la innovación y el mejoramiento de la tecnología involucrada en su desarrollo, se espera traiga consigo mejoras en diferentes ámbitos, tales como:

- Reducción de accidentes al disminuir el error humano en la tarea de conducción
- Mayor eficiencia en el control del tráfico vehicular y por lo tanto menos horas para transportarse
- Mitigación de emisiones de CO2 debido al uso eficiente del combustible
- Eventual disminución del uso de automóviles privados al implementar taxis autónomos de bajo costo
- Aumento en la capacidad disponible de las carreteras permitiendo a los vehículos viajar en distancias más próximas entre ellos
- Permitir el acceso a transporte y movilidad para personas que no pueden conducir por sí mismas

Tal y como se puede observar en los puntos anteriores, los que los vehículos autónomos podrían aportar avances significativos en los siguientes temas: seguridad, confort, ahorro en gastos de transporte y combustible, reducción de emisiones y el mejoramiento en la tarea de conducción [9][10][11]

### ***2.4 Sistemas Avanzados de Asistencia al Conductor (ADAS)***

Los sistemas avanzados de asistencia al conductor (*ADAS* del inglés *Advanced Driver Assistance Systems*), son sistemas electrónicos desarrollados con el propósito de mejorar la seguridad de los pasajeros; así como, mejorar la comodidad en la tarea de conducción, dichos sistemas hacen uso de diferentes sensores y dispositivos periféricos que ayudan a detectar e identificar el medio que los rodea [12][13][14].

Generalmente los sistemas ADAS se pueden clasificar en dos grupos:

- **Pasivos:** Son aquellos sistemas que solamente proporcionan una alerta al conductor para que este reaccione en concordancia a la situación presente. (Ejemplos: advertencia de salida de carril, advertencia de colisión trasera y delantera, etc.)
- **Activos:** Son aquellos sistemas que pueden tomar el control del vehículo con la finalidad de evitar un accidente. (Ejemplos: Control de crucero adaptativo, asistente de mantenimiento de carril, etc.)

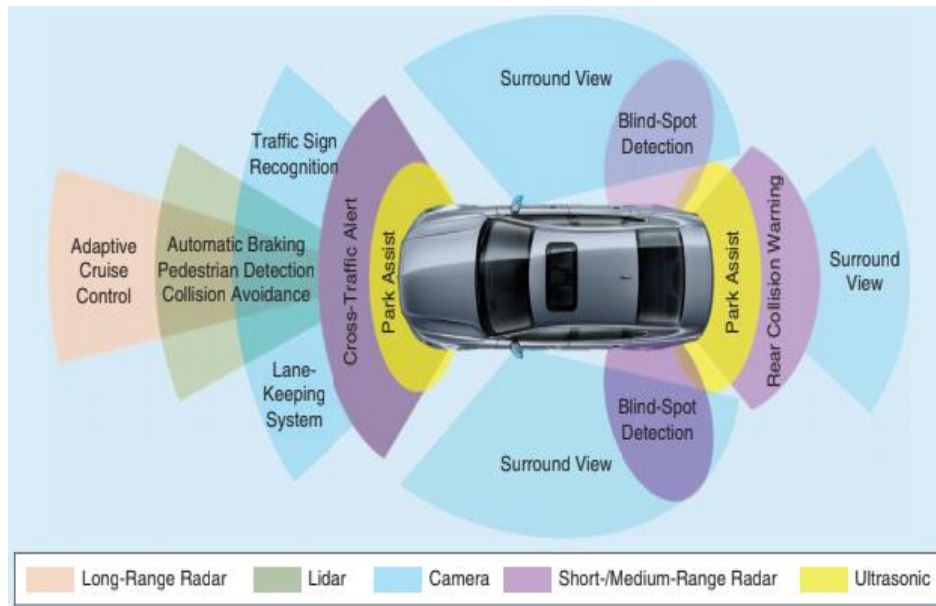


Figura 2.4.1 Representación de los sistemas ADAS de un vehículo [15]

Como se puede observar en la Figura 2.4, hay una gran variedad de sistemas ADAS que tienen el propósito de monitorear el entorno del vehículo en todas direcciones y así realizar acciones pertinentes en situaciones de posible riesgo, los sistemas ADAS se pueden agrupar en cinco categorías principales, sistemas de control lateral, sistemas de control longitudinal, asistentes de estacionamiento, sistemas de visión aumentada, y sistemas de adaptación de velocidad inteligente [16].

### 2.4.1 Sistemas de control lateral

Este tipo de sistemas tiene el propósito de monitorear lateralmente tanto el entorno como la tarea de conducción del vehículo, y realizar una posible acción para prevenir alguna colisión. Dentro de esta categoría podemos encontrar los siguientes sistemas:

- **Sistemas de advertencia de salida de carril y mantenimiento de carril,** estos sistemas monitorean la posición actual del vehículo monitoreando las líneas del carril, lanzando una alerta en el caso del primero y controlando de forma automática la dirección para que el vehículo se mantenga sobre su carril en el segundo caso [16][17]. Este sistema se observa en la imagen 2.4.1.1.

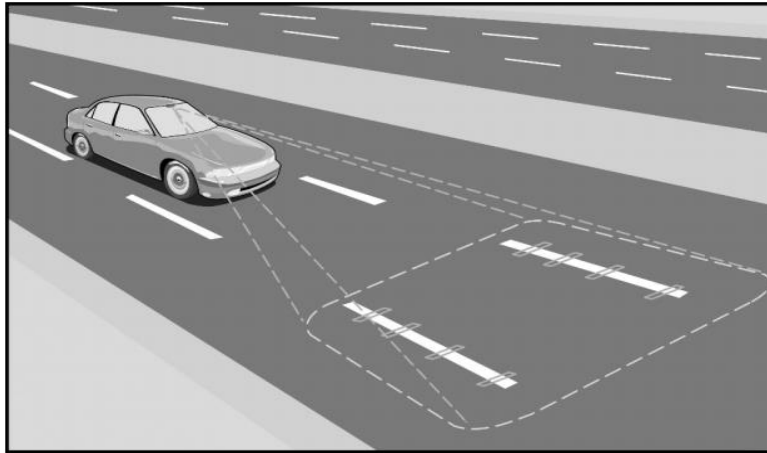


Figura 2.4.1.1. Sistema de advertencia de carril comercial desarrollado por Iteris, Inc. basado en las líneas del carril [17]

- Sistemas de control de estabilidad**, cuya función principal es evitar que el vehículo pierda estabilidad girando o derrapando. Son también conocidos como sistemas de control de giro o sistemas de control de estabilidad electrónicos. Este tipo de sistemas dependen de factores como la velocidad y el coeficiente de fricción que se presente entre la llanta y la autopista, dependiendo si este coeficiente es alto o bajo, la trayectoria del vehículo difiere de cierta manera en específico. En el primer caso el vehículo es capaz de mantener una buena conducción en una autopista curva; sin embargo, en el segundo caso puede, aunado al bajo coeficiente de fricción, una velocidad del vehículo alta, el vehículo puede seguir una trayectoria no deseada de radio mayor. La funcionalidad de los sistemas de control de estabilidad se centra en compensar esa diferencia en la trayectoria no deseada, y aproximarla total o parcialmente al movimiento deseado por el conductor. Tal y como se observa en la figura 2.4.1.2 [17].

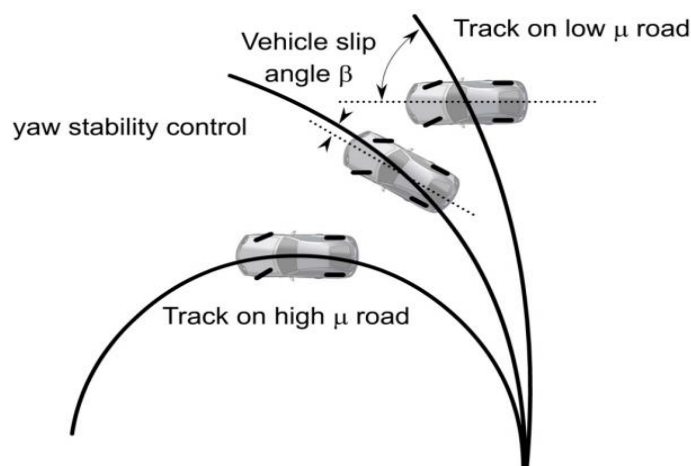


Figura 2.4.1.2. Funcionamiento de los sistemas de control de estabilidad [17]

- **Los sistemas de monitoreo de punto ciego** tienen la función de detectar la presencia de vehículos en los carriles adyacentes del vehículo para alertar al conductor con indicadores de forma visual o audibles. Hacen uso de dos tecnologías principales; la primera de ellas está basada en el radar, en este caso los sensores transmiten y reciben ondas de radio entre el vehículo que posee el sensor y los vehículos adyacentes en la zona del punto ciego; en el segundo caso basado en visión computacional, se hace uso de una cámara digital para verificar el punto ciego y posteriormente procesar las imágenes capturadas e identificar aquellas imágenes con características de un vehículo. Figura 2.4.1.3 [16][18].

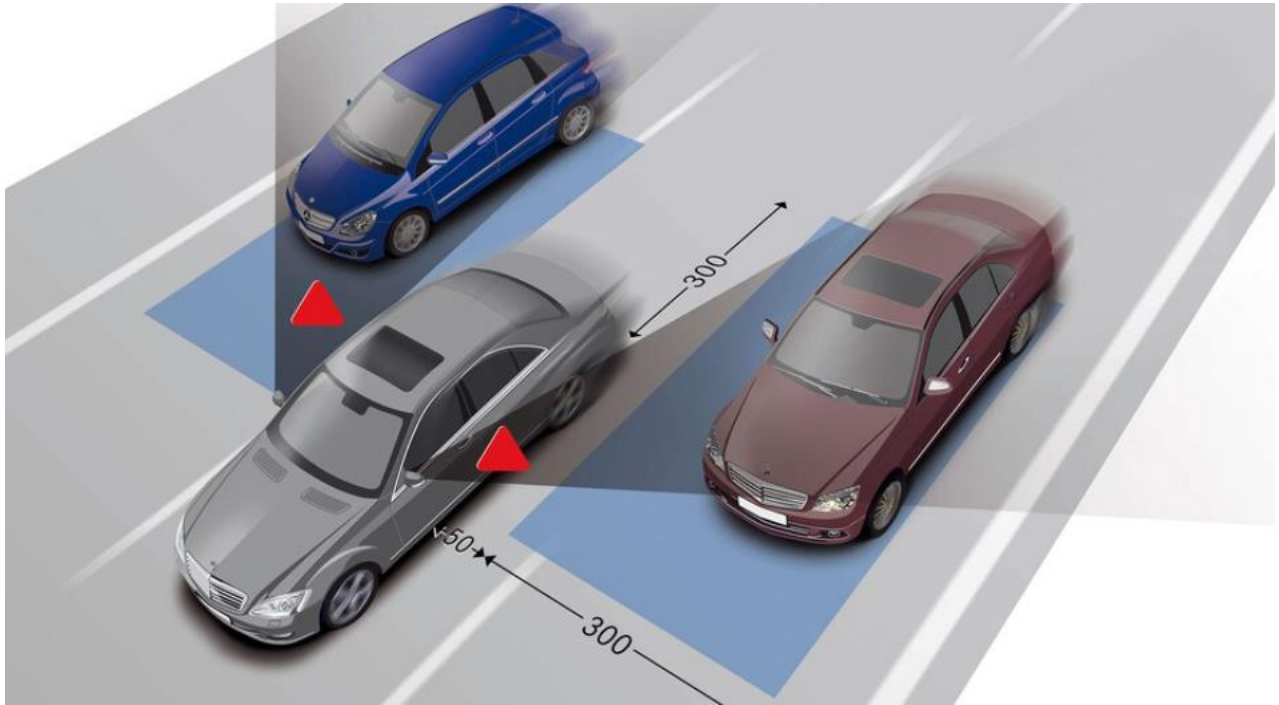


Figura 2.4.1.3. Sistemas de monitoreo de punto ciego [19]

## 2.4.2 Sistemas de control longitudinal

Este tipo de sistemas son aquellos que tienen el propósito de controlar el movimiento longitudinal del vehículo, esto significa que controlan tanto la aceleración, como el frenado del vehículo. Además, implica que se requiere generalmente monitorear la zona delantera y trasera del vehículo para poder accionar algún sistema de control [16][17]. Algunos de los sistemas que se encuentran en esta categoría son:

- **El Control de crucero adaptativo** (*Adaptive Cruise Control*, ACC por sus siglas en inglés) se encarga de mantener una distancia segura entre dos vehículos, dependiendo si la distancia es corta el ACC puede cambiar a un control de espaciado, en el cual el vehículo controla la aceleración y/o el frenado del vehículo. También es posible que no se encuentre ningún

vehículo delante, en estos casos el vehículo viajará a una velocidad determinada, la cual se establece por el usuario, hasta que el ACC determine si el vehículo puede continuar con dicha velocidad o nuevamente otro vehículo se encuentre próximo. Para lograr lo anterior el ACC puede hacer uso de un radar o algún sensor que ayude a cumplir dicho propósito, tal y como se puede observar en la Figura 2.4.2.1.

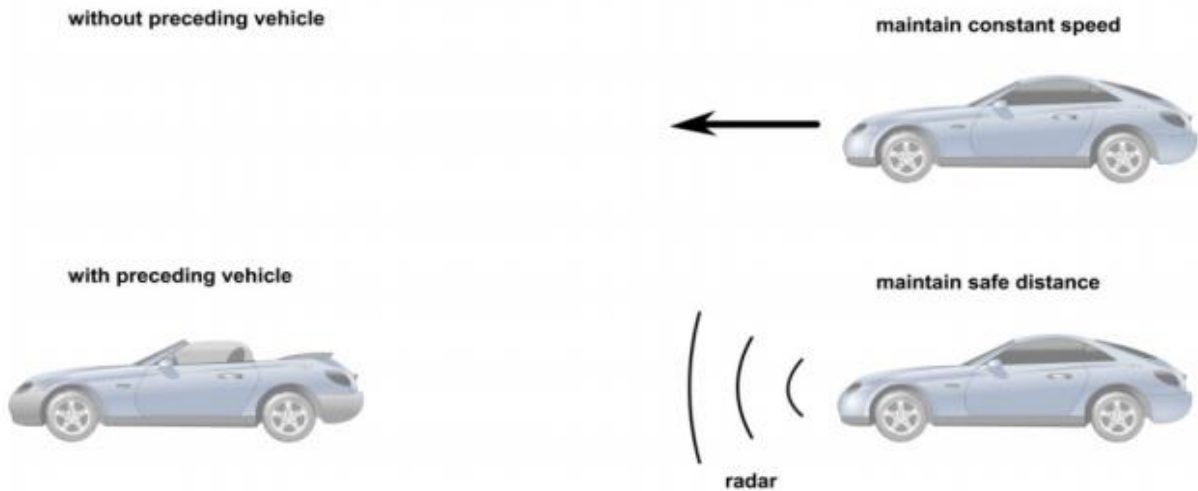


Figura 2.4.2.1. Control de Crucero Adaptativo (ACC) [17]

- **El sistema de evasión de colisión** funciona de manera similar al ACC; sin embargo, la diferencia más notable recae en que el sistema alerta al conductor para reaccionar en caso de un obstáculo en la parte frontal del vehículo, este tipo de sistema es capaz de reducir la velocidad mediante el frenado o desaceleración del vehículo [16][17].
- **El sistema antibloqueo de ruedas** (*Antilock Brake System*, ABS por sus siglas en inglés) tiene el propósito de maximizar la fuerza de frenado y evitar que las llantas se bloqueen durante un frenado fuerte en donde se excede el límite de adhesión de la llanta. Una de las funciones principales del sistema ABS es modular la distribución del frenado de tal forma que la estabilidad, maniobrabilidad y distancia de frenado sean balanceadas adecuadamente, Figura 2.4.2.2 [17].

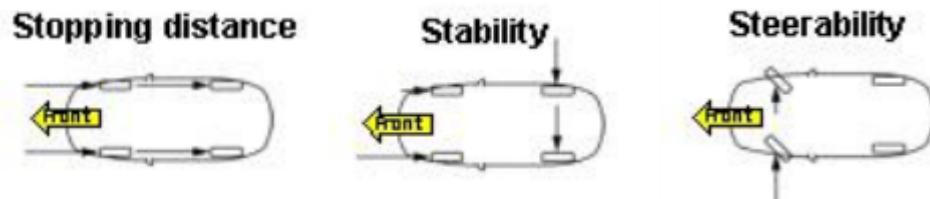


Figura 2.4.2.2. Representación de cómo influye el sistema ABS en la distancia de frenado, estabilidad y maniobrabilidad.

### 2.4.3 Asistentes de estacionado

Los sistemas de ayuda de reversa y estacionado ayudan en la maniobrabilidad del vehículo en bajas velocidades, en este caso el sistema se centra en monitorear el entorno de la parte trasera y delantera.

- **El sistema de ayuda de estacionado** arroja una advertencia al conductor sobre obstáculos mediante el uso de sensores, generalmente ultrasonido, el cual detecta los objetos y determina la distancia a la que se encuentra en su rango de operación, Figura 2.4.3.1 [16][21].

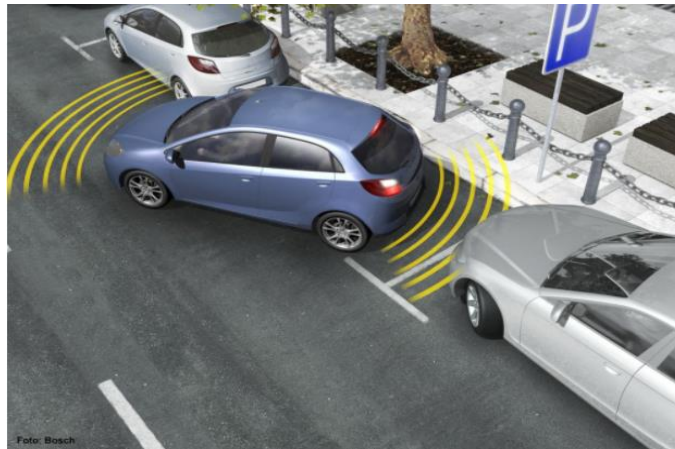


Figura 2.4.3.1. Ejemplificación del funcionamiento de un sistema de ayuda de estacionado [20]

- **Los sistemas de ayuda de reversa** están integrados por una cámara que muestra la parte trasera del vehículo en el panel ubicado en el tablero, de tal forma que los objetos cercanos puedan ser visualizados, generalmente muestran líneas de la trayectoria que el vehículo seguirá e indican un cambio de color cuando esta debe ser modificada, Figura 2.4.3.2 [16][22].



Figura 2.4.3.2. Ejemplificación de un sistema de ayuda de reversa [22]

### 2.4.4 Sistemas de visión aumentada

Los sistemas de visión aumentada son utilizados como asistencia durante condiciones de poca luz o durante situaciones específicas de uso, como puede ser un vehículo aproximándose deslumbrando la visión del conductor ocultando peatones, ciclistas o cualquier objeto potencialmente peligroso. Existen dos tipos de variantes en este tipo de tecnología, las cuales son:

- **Sistemas lejanos basados en infrarrojo (FIR, en inglés *Far Infrared System*)**. Este tipo de sistemas capta la radiación térmica emitida por los objetos en una longitud de onda de infrarrojo de entre 7 y 12  $\mu\text{m}$ , es considerado como un sistema pasivo y es capaz de detectar objetos en rangos de 300 metros y más. Tal y como se puede observar en la Figura 2.4.4.1, en donde se denota la diferencia de temperatura captada por la cámara mediante el brillo de los objetos, por lo tanto, un objeto más brillante tendrá una mayor temperatura [16][23].



Figura 2.4.4.1. Imagen producida por una cámara FIR [23]

- **Sistemas cercanos basados en infrarrojo (NIR, en inglés *Near Infrared System*)**. Este tipo de sistemas hace uso de la emisión de radiación en el rango de 800 nm y 1000 nm, con lo cual se logra que los objetos delante del vehículo puedan ser identificados. Posteriormente, dichos objetos son detectados por una cámara infrarroja y post procesados mediante una unidad de control electrónico (ECU), lo que permite que la detección de peatones se lleve a cabo en rangos de 150 metros o más. Tal y como se observa en la Figura 2.4.4.2, los objetos naturales son reflejados muy cercanamente a la luz visible y es por ello que resulta más familiar visualmente esta representación [16][23].



Figura 2.4.4.2. Imagen producida por una cámara NIR [23]



### 2.4.5 Sistemas de adaptación de velocidad inteligentes

Los sistemas de adaptación de velocidad inteligentes (ISA, *Intelligent Speed Adaptation* en inglés) se basan en el monitoreo de la posición y velocidad de un vehículo, para que se mantengan por debajo del límite permisible de la autopista en cuestión, Figura 2.4.5.1. Esto es posible gracias al uso de sistemas de navegación (GPS), estos pueden ser sistemas pasivos o activos, siendo que algunas de las acciones que realizan varían desde una simple alarma visual al conductor, hasta la aplicación de una fuerza contraria en el pedal de aceleración para disminuir la velocidad o inclusive la regulación del frenado del vehículo de forma automática [16][24].

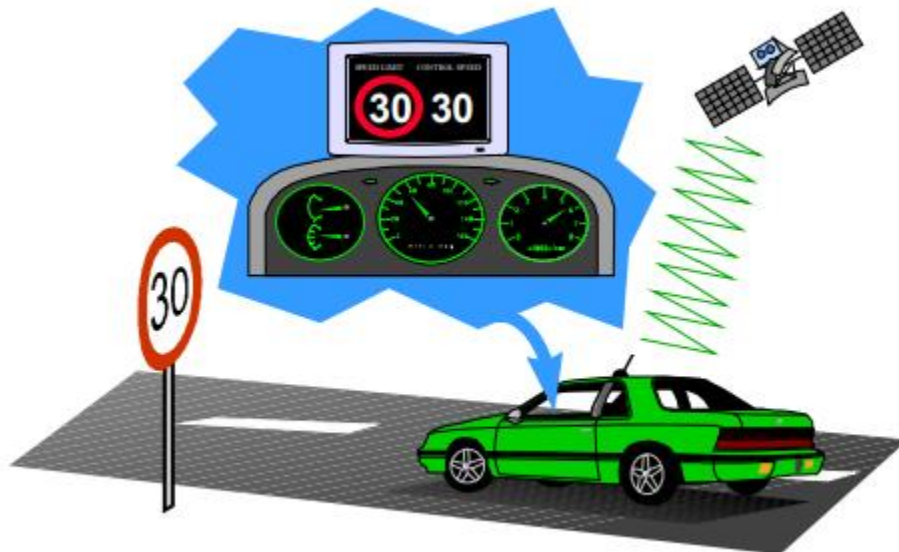


Figura 2.4.5.1. Concepto de los sistemas ISA [25]

### 2.5 Niveles de automatización

La SAE establece que hay seis niveles de automatización, dichos niveles se clasifican de acuerdo a como se lleva a cabo la tarea de conducción, principalmente se agrupa en tres tipos de automatización, que son: nula, media/parcial o completa automatización de un vehículo, de lo anterior se sigue que cada nivel tiene cierto nivel de monitoreo y capacidades para responder a su entorno, tal y como se observa en la Tabla 2.5.1.

Nivel	Nombre	Definición	Control Longitudinal / Lateral	Monitoreo del Ambiente de Conducción	Control de Situaciones de Respaldo/ Emergencias	Capacidad del Sistema/ Modos de Conducción
0	Automatización Nula	La tarea de conducción es llevada a cabo completamente por el piloto en todo momento inclusive en situaciones de emergencia.	Humano	Humano	Humano	n/a
1	Asistencia al Conductor	Adición de funciones de asistencia de manejo, sistemas de control de movimiento lateral (dirección) o longitudinal (aceleración/ desaceleración). El piloto es quien decide cuándo activar dicho control, este sigue supervisando el entorno y manipulando el vehículo de ser necesario.	Humano y Sistema	Humano	Humano	Limitado
2	Automatización Parcial	El vehículo se mueve por sí mismo, control lateral y longitudinal; sin embargo, nuevamente el piloto debe monitorear el entorno para cambiar entre control manual y los sistemas de control.	Sistema	Humano	Humano	Limitado
3	Automatización Condicional	Control lateral y longitudinal, así como capacidad de reconocer sus límites, alerta al conductor en caso de ser necesario. El conductor no requiere de monitorear el ambiente, pero sí de estar presente en caso de que el sistema requiera un cambio manual.	Sistema	Sistema	Humano	Limitado
4	Automatización Elevada	El sistema es capaz de hacer todas las funciones necesarias de manejo en casos específicos de uso, el conductor no realiza ninguna función, ya sea de monitoreo o como respaldo del sistema.	Sistema	Sistema	Sistema	Limitado
5	Automatización Completa	El sistema es capaz de hacer todas las funciones necesarias de manejo en cualquier caso de uso, el conductor no realiza ninguna intervención a lo largo del trayecto del vehículo.	Sistema	Sistema	Sistema	Ilimitado

Tabla 2.5.1. Adaptado de "Definitions for terms related to driving automation systems for on-road motor vehicles", por SAE International, 2016, SAE Standard J3016. ©2016 SAE International [6][8]

Una posible interpretación del impacto que tienen los niveles de automatización en la tarea de conducción de un vehículo, así como algunos sistemas ADAS representativos de cada uno se presenta a continuación. En la Tabla 2.5.2, se encuentran los diferentes sistemas de un vehículo autónomo categorizados directamente en su nivel de automatización correspondiente, así como el impacto que tienen en la tarea de conducción y el escenario real en que se pueden aplicar cada uno de ellos.

Nivel	Sistemas/funciones	Impacto	Escenario
0	Asistente de cambio de carril (LCA), Advertencia de salida de carril (LDW), Advertencia de colisión delantera (FCW) Control de distancia de Estacionamiento (PDC)	<ul style="list-style-type: none"> <li>Mejora en la comodidad de manejo (el conductor debe estar presente en todo momento).</li> <li>Incremento de seguridad y de la eficiencia energética.</li> <li>Mejoramiento del tráfico vehicular (si son sistemas cooperativos).</li> </ul>	Autopistas con acceso limitado, función de asistente en situaciones de embotellamiento.
1	Control de cruceo adaptativo (ACC), Asistente de estacionamiento (PA), Asistente para mantenimiento de carril (LKA)		
2	Asistente de estacionamiento		
3	Chofer en situaciones de embotellamiento, Chofer de autopista (MWC)	<ul style="list-style-type: none"> <li>Comodidad y conveniencia mejoradas (se reduce el costo del tiempo utilizado para transportarse).</li> <li>Seguridad incierta al depender completamente de la habilidad del conductor para reaccionar ante situaciones de emergencia.</li> <li>Incremento en la eficiencia energética y un mejoramiento del tráfico vehicular (si son sistemas cooperativos).</li> </ul>	Chofer en situaciones de embotellamiento y en autopista.
4	Piloto en autopista, estacionamiento automático	<ul style="list-style-type: none"> <li>Comodidad al conducir mejorada considerablemente.</li> <li>Seguridad mejorada a causa de una respuesta automática minimizando el riesgo.</li> <li>Incremento significativo en la eficiencia energética y un mejoramiento del tráfico vehicular (sistemas cooperativos).</li> </ul>	Camiones sobre carriles exclusivos, valet parking automatizado, piloto automático en autopista.
5	Robot taxi (carro privado completamente automatizado)	<ul style="list-style-type: none"> <li>Máxima comodidad y conveniencia</li> <li>Ganancias en la eficiencia energética.</li> <li>Uso de una red de autopistas.</li> </ul>	Servicio electrónico de chofer, servicio de recolección y entrega de bienes automatizado.

Tabla 2.5.2. Escenarios e impacto en la tarea de conducción de algunos sistemas automatizados implementados en los vehículos [10]

## 2.6 Sensores

El funcionamiento de los vehículos autónomos y de sus sistemas dependen de la obtención de información del entorno que los rodea, la manera de conseguirla es mediante el uso de sensores. Al igual que en los sistemas descritos en secciones anteriores, los sensores se pueden clasificar como pasivos o activos; sin embargo, en este caso la definición cambia un poco, es decir,

los sensores pasivos absorben la energía emitida en el ambiente, mientras que los sensores activos emiten energía y miden la señal que es recibida, en estas categorías podemos encontrar los siguientes: sensor ultrasónico, RADAR, LIDAR, cámaras, entre otros.

No solo es necesario conocer qué ocurre en el exterior del vehículo, sino que también se requiere conocer el estado interno del vehículo. Los sensores que cubren dicha necesidad son llamados sensores propioceptivos, algunos ejemplos de estos son: IMU, GNSS, RTK, entre otros [6][26].

En la Figura 2.6.1 podemos observar una forma en cómo se pueden clasificar los principales sensores que son usados para la puesta en operación de los sistemas ADAS.

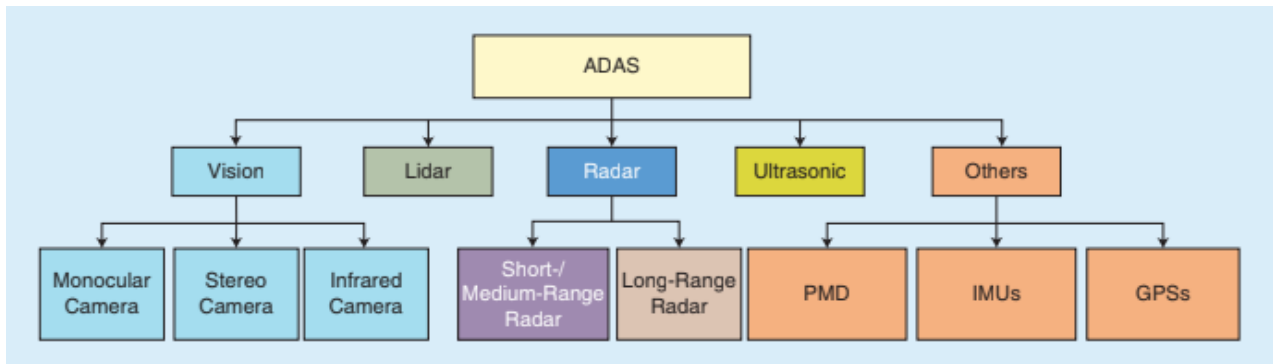


Figura 2.6.1. Principales sensores utilizados para el funcionamiento de los sistemas ADAS [15]

## 2.6.1 Radar

El radar (del inglés *Radio Detection and Ranging*) es uno de los elementos más importantes que utilizan los vehículos autónomos, es usado ampliamente en los sistemas ADAS tales como el monitoreo de punto ciego, el control de crucero adaptativo, el cambio de carril asistido, entre otros. Su principio de funcionamiento se basa en la emisión de ondas electromagnéticas para poder detectar objetos, y consecuentemente poder obtener su posición, distancia próxima, velocidad e inclusive la dirección en la que se encuentra en algunos casos. Hay dos tipos de radar: eco y Doppler. El radar tipo eco emite pulsos que al ser reflejados por los objetos dentro de su rango proveen la información de interés. Por otro lado, el radar Doppler funciona mediante la lectura y análisis de la fase de las ondas emitidas, de esta forma detecta si existe alguna diferencia entre cada una de ellas, el cambio de fase puede ser positivo (cuando se acerca el objeto) o negativo (cuando se aleja el objeto).

Algunos radares tienen la posibilidad de ajustar su ángulo de apertura modificando su rango de alcance dependiendo de la velocidad del vehículo, a baja velocidad se modifica a un ángulo mayor y un rango menor, sucede lo contrario para casos de alta velocidad. Tal y como se muestra en la Figura 2.6.1.1 [6][26].

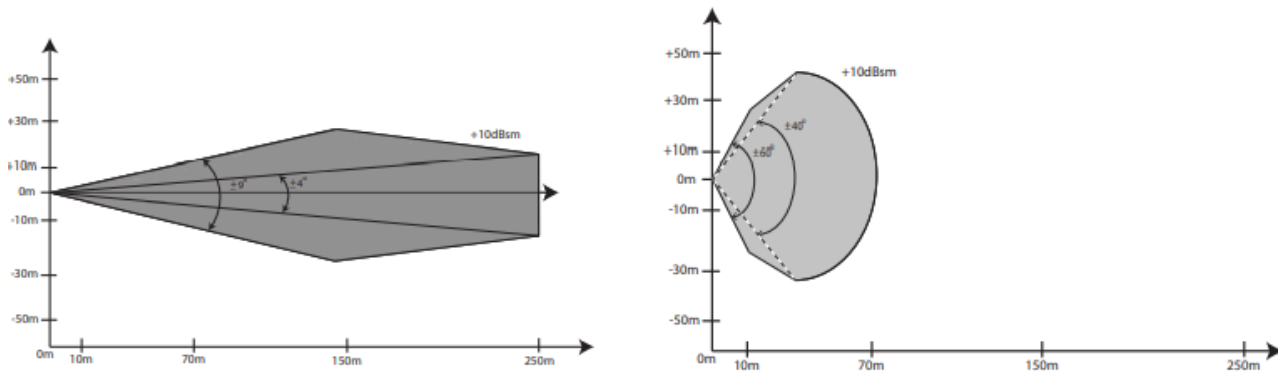


Figura 2.6.1.1. Formas de operación de un radar: a la izquierda funcionamiento de largo alcance, a la derecha de corto alcance [6]

El uso del radar representa algunas ventajas en los sistemas ADAS, ya que pueden ser utilizados en condiciones de poca luz y en cualquier condición climática. Además, puede mejorar efectivamente la seguridad del vehículo. Por otra parte, el radar tiene un efecto pobre sobre objetivos no metálicos y en ocasiones puede llegar a generar falsos positivos producidos por el rebote de la señal emitida [6][26].

## 2.6.2 Sensor Ultrasónico

El sensor ultrasónico utiliza ondas de sonido de baja frecuencia, del orden de 20 kHz a 40 kHz, con el objetivo de medir la distancia que existe a un objetivo, para ello hace uso del principio de tiempo de vuelo (ToF por sus siglas en inglés), este consiste en medir el tiempo que tarda en recibirse el eco que ha sido emitido por el sensor, tal y como se observa en la Figura 2.6.2.1 [6][26]. Esta diferencia se puede medir de dos maneras, la primera de ellas es de forma directa, la cual funciona como se ha descrito anteriormente; la segunda es de forma indirecta, en donde se envía una onda sinusoidal de manera continua y se mide la diferencia en la fase de la señal recibida [27]

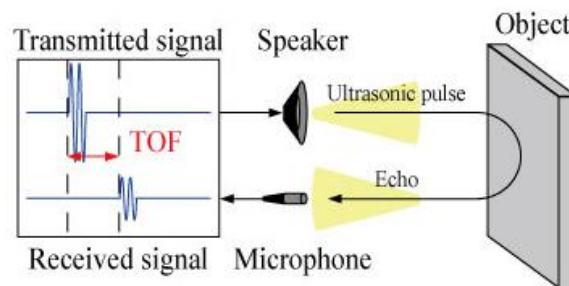


Figura 2.6.2.1. Principio de funcionamiento del tiempo de vuelo (ToF) [28]

Este tipo de sensores se utilizan principalmente en los asistentes de estacionamiento de los vehículos, debido a que son relativamente asequibles y cumplen su propósito adecuadamente.

Se desenvuelven bien en la mayoría de condiciones climáticas (neblina, lluvia, nieve o hielo), no son dependientes de la luz y pueden detectar objetos no metálicos. Algunas desventajas de estos sensores incluyen una baja resolución y rango de operación. También, pueden ser alterados por ruidos externos del entorno y provocar falsos positivos debido al rebote de la señal [6][26].

### 2.6.3 LiDAR

Lidar es un acrónimo del inglés *light detection and ranging*. Su principio de funcionamiento es muy similar al del radar; sin embargo, el sensor lidar hace uso de ondas de luz en lugar de sonido, generalmente se compone de diodos láser (emisor), fotodiodos (receptor) y un espejo con el que se dirige el rayo láser en dirección vertical u horizontal. Son usados ampliamente en diferentes aplicaciones por su capacidad de mapear el entorno ya sea en 2D o 3D dependiendo del tipo de sensor utilizado. También, se pueden clasificar dependiendo de su construcción en rotatorios o de estado sólido. El sensor lidar 3D permite mapear con precisión el entorno dependiendo del número de láseres equipados en él, y dependiendo del número de canales pueden ser utilizados en sistemas ADAS, tales como el control de crucero adaptativo y la evasión/reconocimiento de objetivo [6][26]. Por otro lado, el sensor lidar de estado sólido permite realizar las mismas funciones sin la necesidad de tener partes móviles, esto se logra gracias a la implementación de un micro espejo MEMS (del inglés *Micro Electro Mechanical Mirrors*), el cual se sincroniza con el láser emitido para escanear el campo de visión horizontal y adicionalmente mediante una lente difusora proyectar los puntos de manera vertical y poder capturar la señal emitida en algún fotoreceptor, Figura 2.6.3.1 [26].

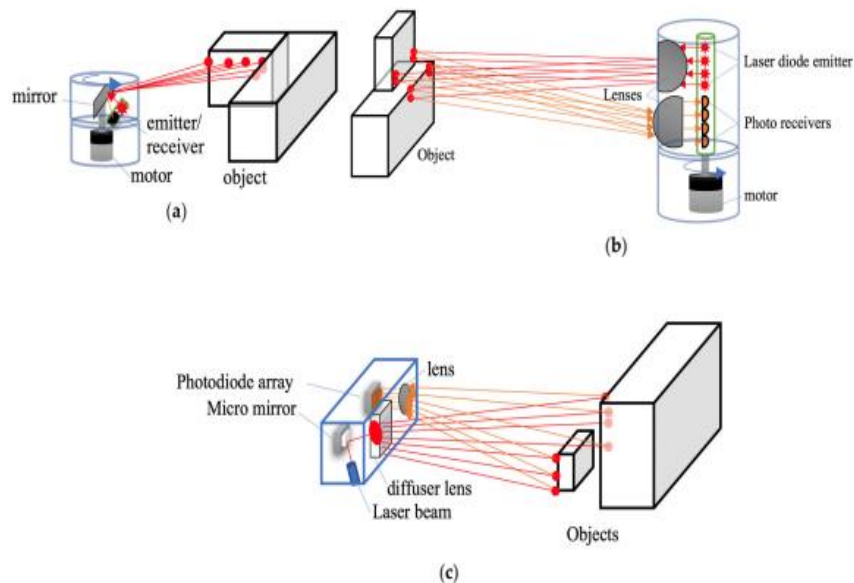


Figura 2.6.3.1. a) lidar 2D rotatorio, b) lidar 3D rotatorio, C) lidar 3D de estado sólido

Los sensores lidar tienen una serie de ventajas que lo han convertido en la elección principal si se trata de realizar mapeo del entorno en 3D, entre ellas podemos mencionar su capacidad de funcionar en el día y noche, así como el papel que realiza en el posicionamiento

interno del vehículo y en áreas donde no están disponibles los sistemas de posicionamiento global. Por otro lado, se puede tener una desventaja si hablamos del costo requerido para su producción en masa equipado en vehículos; sin embargo, este problema se está tratando de solventar con los sensores de estado sólido. Otra desventaja importante es que estos sensores son sensibles en climas de lluvia, nieve, neblina o polvo, lo que provoca que se genere ruido en la señal recibida [26][6].

### **2.6.4 Cámaras**

Las cámaras son los sensores de visión más comúnmente utilizados. Los sistemas ADAS utilizan una o más cámaras para capturar, analizar y procesar imágenes para reconocimiento de objetos. Una de las aplicaciones más importantes de las cámaras se centra en la visión trasera para los asistentes de estacionamiento; sin embargo, también son utilizadas en diversos sistemas como la advertencia de salida de carril, el monitoreo de interiores y exteriores, así como realidad aumentada y autopilotaje. Por lo general se ubican en la parte frontal del retrovisor tal y como se observa en la Figura 2.6.4.1. Existen diferentes tipos de cámaras, estas pueden ser monoculares, estéreo, de infrarrojo o basadas en el principio de ToF. La mayoría de ellas son consideradas sensores pasivos a excepción de la cámara de ToF, que funciona bajo el mismo principio de un radar, lidar o sensor ultrasónico [6][15][26].

- **Cámaras monoculares:** como su nombre lo indica solo tienen un lente, requieren de un bajo procesamiento de imagen ya que solo tienen un punto de visión durante todo momento, generalmente son usadas en la detección de obstáculos, peatones, carriles y señales de tráfico. Sin embargo, este tipo de cámaras no son confiables en la estimación de distancias, ya que no proveen información sobre la profundidad [15].
- **Cámaras estéreo:** constan de dos lentes separadas por una distancia y mirando en la misma dirección, su principal función es recabar información en 3D del entorno estimando la profundidad, esto es posible debido a que se concatenan las imágenes de ambas cámaras y se realiza el cálculo de disparidad entre ambos lentes [6][15][26].
- **Cámaras infrarrojas:** son sensores pasivos que trabajan con ondas del orden de 780 nm a 1 mm, generalmente son utilizadas en situaciones con umbrales de iluminación; por ejemplo, el deslumbramiento debido al sol; y en la detección de cuerpos calientes de cualquier material, como peatones, animales y otros vehículos [15][26].
- **Cámaras ToF:** son sensores activos capaces de capturar completamente un escenario determinado en 3D, compuestos generalmente por un dispositivo mezclador fotónico utilizado para calcular la distancia de la señal [6][26].



Figura 2.6.4.1. Cámara montada en el espejo retrovisor [15]

En general los diferentes tipos de cámaras son útiles dado que pueden capturar una gama más amplia de frecuencias, incluido el color de los objetos, esto permite tener una mejor interpretación visual del entorno y comparadas con el radar y el lidar tienden a tener un costo menor. Por otro lado, las cámaras son sensibles a los cambios climáticos y a la luz del entorno.

### **2.6.5 GNSS**

Los sistemas globales de navegación por satélite (GNSS por sus siglas en inglés) son ampliamente utilizados para el posicionamiento en diversas aplicaciones terrestres, marítimas y aéreas. Su principio de funcionamiento consiste en proporcionar una posición absoluta de un receptor con respecto a un punto fijo, para llevar a cabo esto es necesaria la intervención de por lo menos 24 satélites orbitando en la atmósfera transmitiendo una señal de regreso a la tierra con información sobre su posición, velocidad, tiempo, etc. En el caso del sistema de posicionamiento global (GPS por sus siglas en inglés) generalmente se necesitan 4 satélites para poder obtener una lectura confiable de la posición.

La posición de un receptor GNSS se obtiene mediante el principio de trilateración. Este principio funciona como su nombre lo indica con tres señales, primeramente, se realiza el cálculo de la diferencia de tiempo entre que cada señal es emitida por los satélites y recibida por el receptor en la tierra (propagación de la señal), posteriormente se calcula la distancia multiplicando el tiempo por la velocidad de propagación. Para determinar la posición del receptor GNSS se requiere conocer la posición y distancia a cada uno de los satélites, el lugar donde interceptan las 3 señales será la posición del receptor. Figura 2.6.5.1 [6][26].



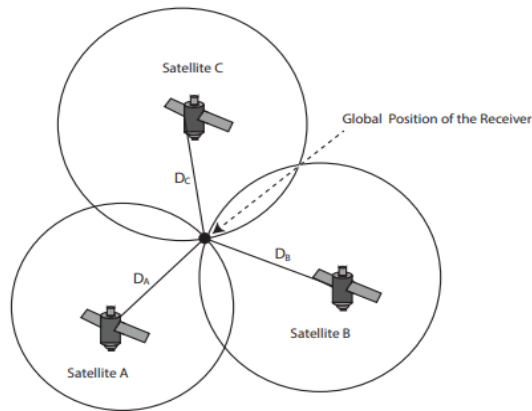


Figura 2.6.5.1. Principio de trilateración para determinar la posición de un receptor GNSS [6]

Los sistemas GNSS poseen la gran ventaja de tener cobertura alrededor de cualquier lugar en la tierra y bajo las condiciones adecuadas pueden determinar de manera correcta la posición. De igual forma, no es afectado por errores acumulativos en la señal, y se han vuelto una opción asequible y disponible incluso en muchos dispositivos actuales. Sin embargo, debido a que depende de las señales satelitales, estas pueden ser afectadas por el ruido de la señal en los receptores o por la reflexión de las mismas en diferentes objetos, por tanto, requiere de una línea clara entre emisor y receptor para operar óptimamente [6][26].

### 2.6.6 IMU

Unidad de medición inercial (IMU por sus siglas en inglés), es un dispositivo conformado por una serie de giroscopios y acelerómetros que sirven para medir aceleración lateral y velocidad angular respectivamente. El propósito principal de una IMU es la obtención de la velocidad, posición y orientación estimadas. Generalmente, están compuestas por tres acelerómetros y tres giroscopios, que en combinación hacen que sea posible medir hasta seis grados de libertad, es decir, medir la aceleración en las coordenadas (X,Y,Z); así como, la rotación sobre cada eje (yaw, pitch, roll); por otro lado, también es posible encontrar IMU con hasta 9 grados de libertad, para tal propósito se integran los magnetómetros, estos son sensores capaces de determinar la orientación basados en fuerza y dirección del campo magnético de la tierra, los nueve grados de libertad mencionados se ilustran en la Figura 2.6.6.1.

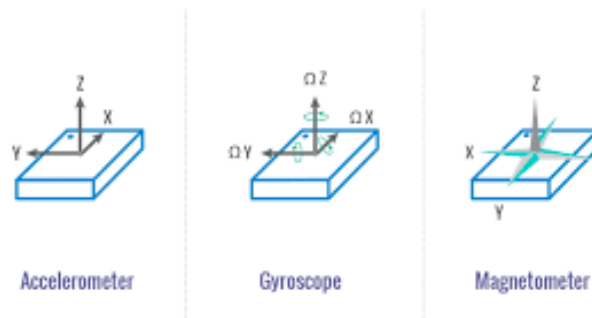


Figura 2.6.6.1. Representación de los movimientos estimados por una IMU de nueve grados de libertad [29]

El uso e implementación de las IMU resulta muy favorable en los vehículos autónomos, ya que tienen una alta disponibilidad y dependen principalmente de las condiciones iniciales, esto quiere decir, que el entorno y sus factores externos no afectaran en gran medida la estimación de la información. Por otro lado, es un sensor que tiende a presentar errores como el ruido, el offset y el factor de escala [6][30].

### 2.6.7 Odómetros

Los odómetros son una parte fundamental para la estimación del estado actual del vehículo, su principal función es medir la distancia recorrida por el vehículo, esto se realiza mediante el cálculo de la multiplicación del número de rotaciones de la llanta por el diámetro del neumático. Existen dos tipos de odómetros, pueden ser considerado como sensores pasivos o activos.

En el caso particular de la industria automotriz son utilizados para determinar la velocidad actual del vehículo, así como la distancia recorrida por cada llanta. Lo anterior resulta de utilidad para los sistemas ADAS y algunos sistemas de seguridad como los frenos ABS.

Por lo general se utiliza un sensor de velocidad acoplado directamente a la llanta como elemento pasivo y un sensor de efecto hall o sensor óptico en el caso de ser un elemento activo; sin embargo, estos últimos requieren de una fuente de energía externa para poder funcionar. El principio de funcionamiento en los tres casos es muy similar, se basa en la fluctuación del elemento en cuestión con respecto a una referencia fija, esta puede ser una variación de voltaje debida a la fluctuación del mismo en el sensor de velocidad (Figura 2.6.7.1) o bien una variación del campo magnético o de a señal adquirida por el sensor en cuestión (Figura 2.6.7.2).

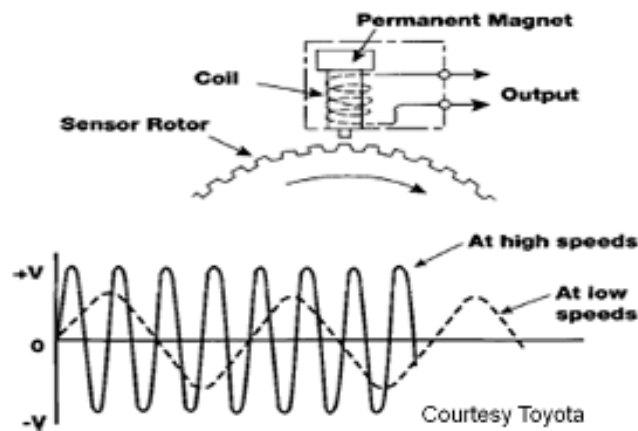


Figura 2.6.7.1 Principio de funcionamiento de un sensor de velocidad pasivo [31]

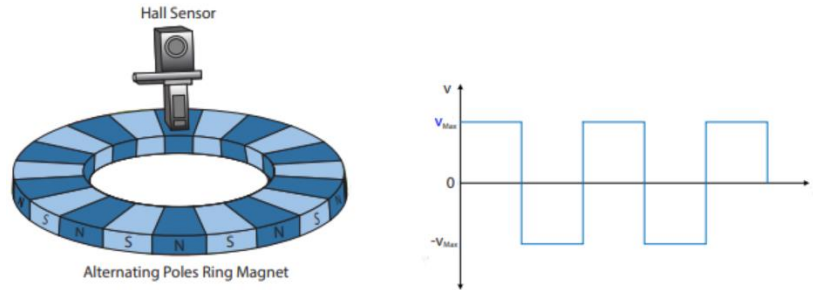


Figura 2.6.7.2 Principio de funcionamiento de un sensor de velocidad activo, sensor de efecto Hall [6]

Este tipo de sensores resultan ser costeables y proveen información acertada sobre la distancia recorrida de un vehículo; no obstante, no están exentos de presentar un error en la medición de la señal, esto debido principalmente a la acumulación de errores de los factores externos, como el deslizamiento de la llanta, superficies desniveladas, etc. Por tal razón son utilizados en conjunto con los sensores previamente mencionados, tales como las IMU y los GNSS [6][32].

## 2.7 Ejemplos de vehículos autónomos

En la actualidad hay una gran variedad de empresas activas en el desarrollo de tecnología e implementación de los vehículos autónomos, existen algunas que llevan la delantera; sin embargo, existe una especie de consenso en la comunidad sobre cuáles son las tres empresas que lideran el avance tecnológico en este campo, estas se enlistan a continuación [33].

- Waymo
- GM cruise
- Argo AI

### 2.7.1 Waymo

Inicialmente conocida como "proyecto de vehículo autónomo de Google" inicializado en el año 2009, Waymo es una empresa cuyo propósito principal es el desarrollo de la tecnología utilizada en vehículos autónomos, es una de las empresas con mayor tiempo y desarrollo de investigación en este ámbito, oficialmente. Actualmente Waymo se encuentra aliada con grandes empresas automotrices como Jaguar Land Rover, alianza Renault-Nissan-Mitsubishi y Volvo Car Group, entre otras.

Su principio de diseño se centra en construir y diseñar un "conductor" seguro, precavido y eficiente en lugar de un construir un vehículo en sí, esto se debe a que su enfoque está principalmente relacionado con el desarrollo de un sistema de navegación autónoma, es decir, la interacción de los diferentes sensores empleados como un todo. La principal meta de Waymo es

operar y realizar las tareas dinámicas de un vehículo completamente autónomo, sin la necesidad de un conductor en ningún momento [33][34].

Waymo se ha concentrado en dos diferentes líneas de implementación, *Waymo one* y *Waymo via*. La primera de ellas tiene su foco de aplicación en el traslado de personas, es decir, un servicio de taxi autónomo o de uso particular en las ciudades que así lo permitan. La segunda de ellas, tiene un enfoque particular en la transportación y entrega de materiales y bienes comerciales en general. Es por ello que cada una de ellas tiene su particular diseño y objetivo, ambas ramas de vehículo se pueden observar con mayor detalle en la Figura 2.7.1.1.



Figura 2.7.1.1. Waymo One y Waymo Driver [35]

Tal y como se describió en secciones anteriores, los vehículos autónomos hacen uso de una gran variedad de sensores, en este caso particular en la Figura 2.7.1.2 podemos observar el acomodo de los mismos en un vehículo de la línea Waymo One, específicamente representado en un vehículo de la marca Jaguar modelo I-Pace 2021. Cabe destacar que los sensores utilizados en puntos frontales, laterales y posteriores del vehículo son exactamente los mismos, sensores lidar, cámaras y radares, cuyo propósito es la obtención de datos del entorno, es decir, una perspectiva en 360° del vehículo.

Adicionalmente, el Lidar central en la parte más alta del vehículo tiene un alcance de hasta 300 metros y es complementado con cuatro más en los puntos periféricos del vehículo. El sistema de cámaras está compuesto por 29 elementos, su propósito es proporcionar imágenes con mayor resolución y rango para procesar una perspectiva mayor del entorno. En cuanto al sistema de radares, esta conformado por seis elementos situados en la periferia del vehículo que puede seguir el estado de objetos estáticos y en movimiento a mayores distancias.



Figura 2.7.1.2. Waymo one, I-Pace 2021 Jaguar Land Rover (JLR) [35]

Así mismo, el proceso de diseño que llevan a cabo se basa en adaptar a los vehículos de las empresas aliadas con su tecnología denominada de “conductor” para producir vehículos completamente autónomos [34][35].

## 2.7.2 GM Cruise

Fundada en el año 2013, Cruise es una de las empresas mejor posicionadas en el desarrollo de vehículos autónomos, con apoyo de grandes empresas como Honda, GM y SoftBank, entre otras [33][36].

Su principio de diseño se basa en asegurar la seguridad de los pasajeros, dentro de sus desarrollos principales, actualmente se encuentran dos proyectos, “*Our First-Gen AV*” y “*The Cruise Origin*”.

El proyecto “*Our First-Gen Av*” hace uso de un vehículo modelo Chevy Bolt para su desarrollo modificado con características propias de un vehículo autónomo, dicho vehículo cuenta con un cuarenta por ciento de Hardware dedicado exclusivamente a las funciones de navegación autónoma. Adicionalmente, tiene un monitoreo de 360° sobre su entorno para monitorear la información a su alrededor, como peatones, construcciones, carros, bicis y condiciones del camino. Esta versión del vehículo resulta ser muy similar a la estructura de la mayoría de vehículos autónomos que están en desarrollo, teniendo en la parte alta el sensor lidar y radar principal, interconectados mediante la propia estructura del vehículo tal y como se puede observar en la Figura 2.7.2.1 [36].

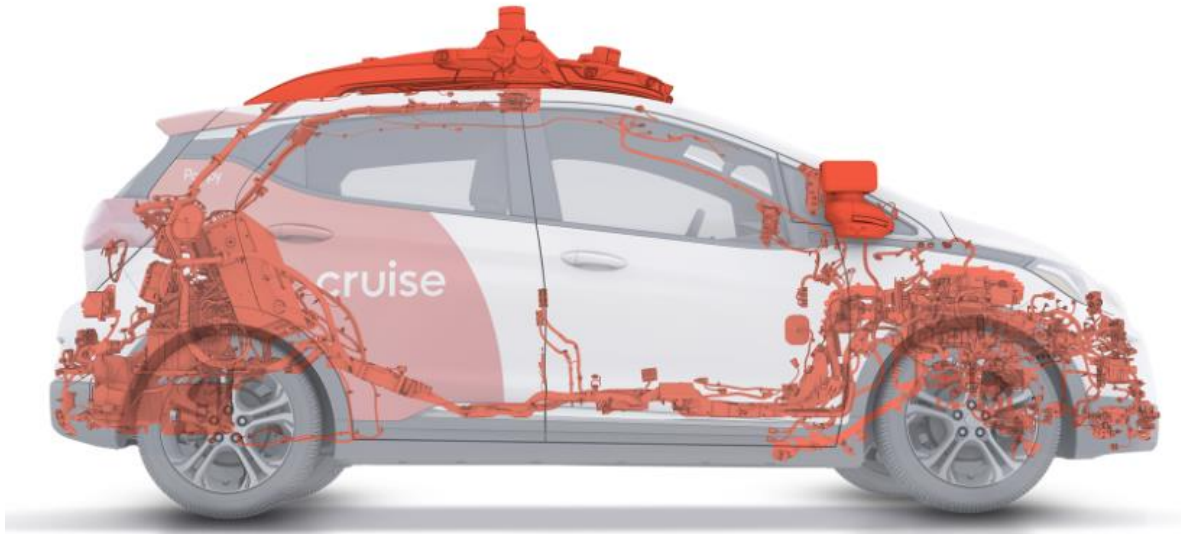


Figura 2.7.2.1. First-Gen AV, Chevrolet Bolt. AV Hardware [36]

Recientemente en el año 2020 la empresa presentó un vehículo aproximadamente del tamaño de una SUV llamado *"The Cruise Origin"* como una nueva propuesta de movilidad en la que no hay un volante, espejos, pedales o elementos propios requeridos por una persona para conducir, es decir, centrándose en la experiencia del pasajero y no del conductor. Esta es una propuesta muy interesante, ya que como se mencionó su enfoque y diseño centraliza su atención en el espacio útil dentro del vehículo, tal y como se puede observar en la Figura 2.7.2.2, no presenta ninguna cajuela en la que se deba revisar el motor integrado, también el sistema de puertas propuesto deja un gran espacio para el acceso del usuario. Tal y como el resto de vehículos autónomos, depende en gran medida de los sensores utilizados, entre los que destacan sensores lidar, radares y cámaras dentro de su estructura [36][37].



Figura 2.7.2.2. The Cruise Origin [36]

### 2.7.3 Argo AI

Es una compañía independiente fundada en 2017, en colaboración con capital e inversiones de grandes empresas automotrices como Ford y Volkswagen (más recientemente). Su principio fundamental de diseño está basado en cerciorarse de la seguridad del uso de sus sistemas, software y hardware una vez son implementados en vehículos ya existentes, en este caso, vehículos de sus aliados. Para lograr lo anterior, construyen y diseñan sus sistemas con base en regulaciones de seguridad estándar, tales como; la ISO 26262, cuyo propósito es determinar requerimientos para el software y hardware del sistema eléctrico y electrónico; adicionalmente, la ISO 21448 que define los estándares de seguridad para las funciones de asistencia al conductor, así como las funciones propias de un vehículo autónomo [38].

Argo ha desarrollado su propio sistema “Self-Driving System” (SDS), este permite al vehículo en cuestión operar de manera autónoma con ciertas restricciones geográficas y dependiendo del clima, cumpliendo la definición de nivel 4 de automatización propuesta por SAE International. En la Figura 2.7.3.1 se muestra la arquitectura de construcción e implementación del SDS.

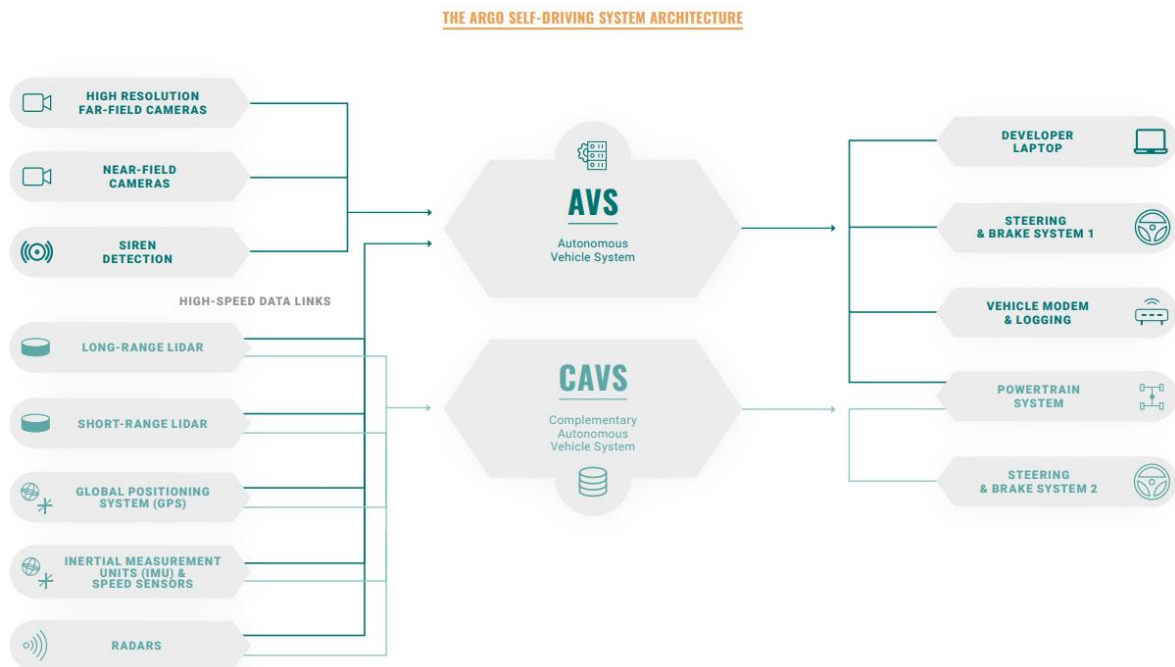


Figura 2.7.3.1. Arquitectura del Sistema SDS [38]

Como elementos destacados se encuentran el funcionamiento del llamado “Autonomous Vehicle System” (AVS) y del “Complementary Autonomous Vehicle System” (CAVS) de la arquitectura del SDS. El primero se considera como la computadora principal del sistema, cuya función primordial es monitorear en todo momento los alrededores del vehículo (360°) y está

diseñada para controlarlo de manera segura ante todo tipo de escenarios posibles con la finalidad de evitar una colisión; adicionalmente, la computadora del AVS realiza acciones como: adquisición de datos, localización, mapeo y planeación de movimiento del vehículo. Por otro lado, La computadora del CAVS corre en paralelo al AVS, siendo un sistema de respaldo en caso de que este entre en un estado de degradación o deje de comunicarse, en otras palabras, provee una capa extra de seguridad para realizar determinadas acciones de emergencia y evitar una colisión.

Dicho sistema tiene una distribución particular en el vehículo, en este caso la Figura 2.7.3.2 muestra el acomodo del sistema SDS equipado en el modelo híbrido de la Ford Escape.

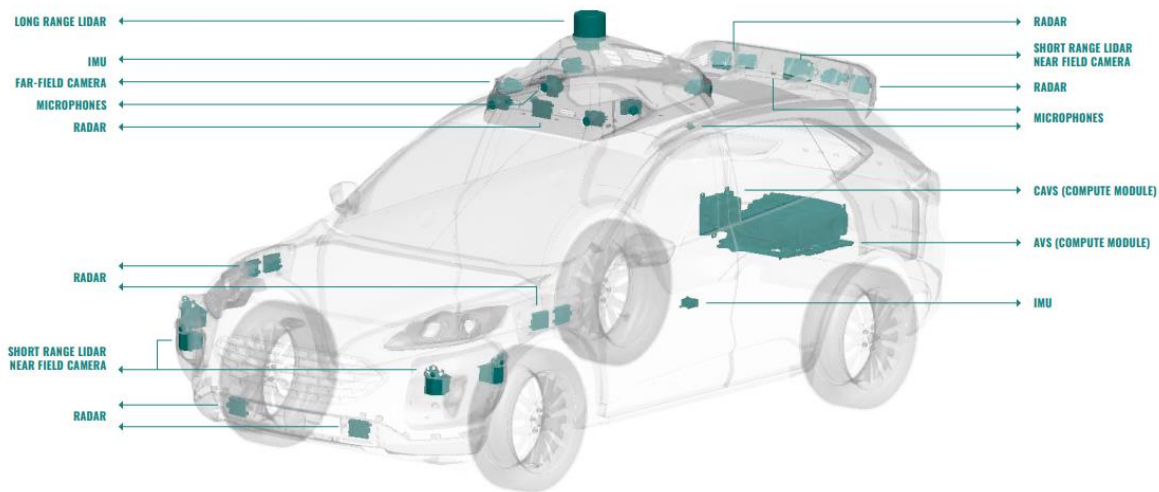


Figura 2.7.3.2 Equipamiento del SDS de Argo en un vehículo híbrido de prueba, modelo Ford Escape [38]

Tal y como se puede observar, la distribución de los sensores del SDS continua con el mismo patrón que la mayoría de vehículos autónomos en desarrollo, variando ligeramente en la parte delantera con un acomodo de los sensores frontales cerca de las luces del vehículo en lugar de los espejos laterales [38].

## 2.8 Conclusión

A lo largo de este capítulo se ha revisado la evolución de los vehículos autónomos, es visible la transformación que ha ocurrido en los sistemas que lo componen, no solo se han vuelto más pequeños, sino también más precisos. Queda claro que son máquinas repletas de diversos sistemas, algunos propios del vehículo como los conocemos y otros dedicados exclusivamente a la implementación de un vehículo autónomo. Además, para que estos puedan funcionar y desarrollar adecuadamente su tarea de acuerdo al nivel de automatización en el que se encuentren, deben interactuar en conjunto como un todo. Esto se reafirma en todo momento, ya que las compañías que actualmente desarrollan e invierten recursos en este tipo de tecnología, basan su principio de diseño en hacer un vehículo que funcione como un todo y satisfaga los requerimientos necesarios para su implementación a nivel comercial. En definitiva, los vehículos



autónomos tienen mucho por desarrollar, pero a su vez, tienen una fuerte propuesta de movilidad que está siendo empujada cada vez con mayor fuerza en nuestro mundo actual, probablemente es cuestión de un par de años para poder ver este tipo de vehículos abiertamente en el mercado.



## 3. Simuladores

De manera general se puede representar un sistema de navegación autónomo tal y como se observa en la Figura 3.1. Este tipo de sistema es puesto en operación gracias a diversos factores, entre los que destacan el uso de simuladores; así como, diversas pruebas de software/hardware y validación de los cada uno de ellos.

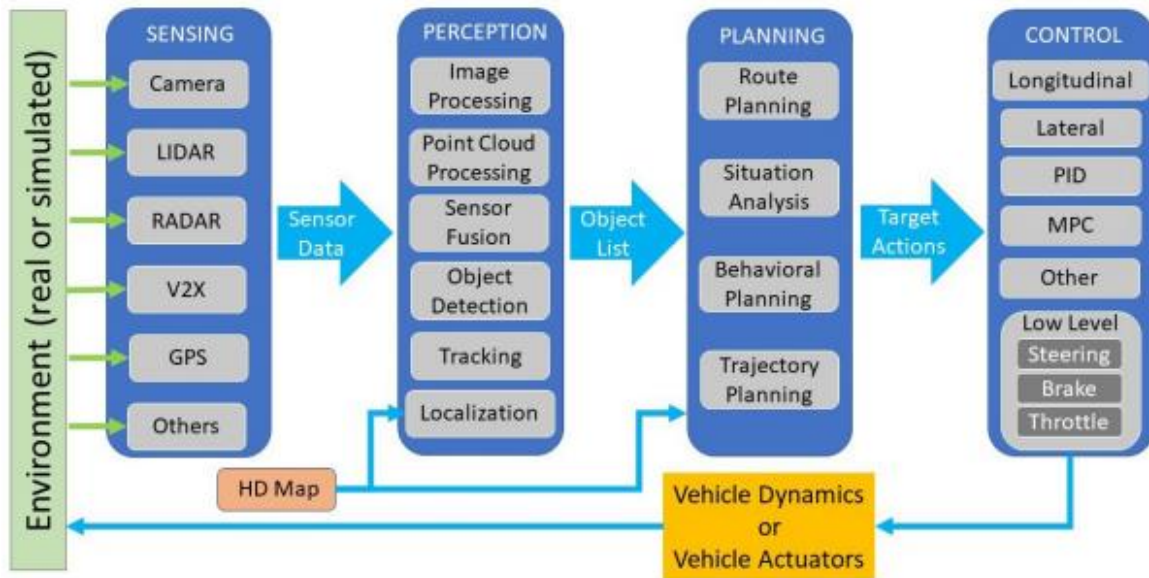


Figura 3.1. Sistema de navegación autónomo genérico [39]

### 3.1 ¿Qué es un simulador?

De acuerdo a la definición propuesta por *Oxford Languages* un simulador se define como:

*“Dispositivo o aparato que simula un fenómeno, el funcionamiento real de otro aparato o dispositivo o las condiciones de entorno a las que están sometidos una máquina, aparato o material.”*

En este caso, el objetivo principal de los simuladores es entrenar a los vehículos autónomos (VA), para llevarlo a cabo se recrean de manera digital los componentes de los VA, tanto software como hardware. De la misma manera se generan el entorno, escenarios y fenómenos físicos relacionados de la vida real en el que estos se desempeñan. Los simuladores permiten probar la eficacia de un VA y lograr optimizar sus sistemas [40].

### 3.2 Campos de aplicación de los simuladores

Gracias al desarrollo de los simuladores de VA y la relevancia que toman para mejorar la tarea de conducción se han expandido a diferentes campos de aplicación, entre los cuales destacan:

- **Diseño y mejoramiento de sistemas en el vehículo:** en este caso la simulación es una herramienta muy útil capaz de ahorrar tiempo en la realización física y validación de los sistemas, algunos ejemplos son los sistemas ADAS, componentes eléctricos como los Faros, entre otros.
- **Entrenamiento:** simulaciones en tiempo real para el acondicionamiento, educación vial y entrenamiento del personal encargado de manejar algún vehículo, ya sean carros, autobuses o camiones, etc.
- **Prueba de nuevas estructuras:** este tipo de simulaciones está enfocado en el análisis del desarrollo del tráfico con el propósito de optimizar recursos para la construcción de instalaciones de transporte.
- **Seguridad y entorno:** otra aplicación es la implementación de carreteras inteligentes y estudios sobre el comportamiento de los conductores ante situaciones de peligro, reconstrucción de accidentes y evaluación del impacto de las emisiones contaminantes.
- **Investigación:** simulaciones de tráfico con el propósito de mejorar el flujo del mismo mediante el apoyo de estudios matemáticos y estadísticos. Lo cual resulta en la investigación de elementos de control del entorno, tales como estrategias de control de señales y sistemas avanzados de control del tráfico [40]

### ***3.3 Métodos de simulación***

El proceso de validación de un VA y sus sistemas que conlleva tanto la simulación del sistema como un proceso de pruebas físicas en un ambiente controlado; así como, aquellas que son desarrolladas en el escenario real de operación, estas son las llamadas pruebas “*test track*” y “*On Road*” respectivamente. En la Figura 3.3.1 se representa de manera general la integración de un simulador durante un proceso de validación, se observa que su papel toma relevancia antes de ser llevado a dichas pruebas. Como cualquier herramienta, los simuladores requieren de parámetros e información para procesar, esta información puede ser provista de diferentes fuentes de información como: bases de datos sobre sensores y pruebas en pista registradas con anterioridad; complementariamente, se requiere identificar los elementos estáticos y dinámicos de cada uno de los escenarios de interés [39].

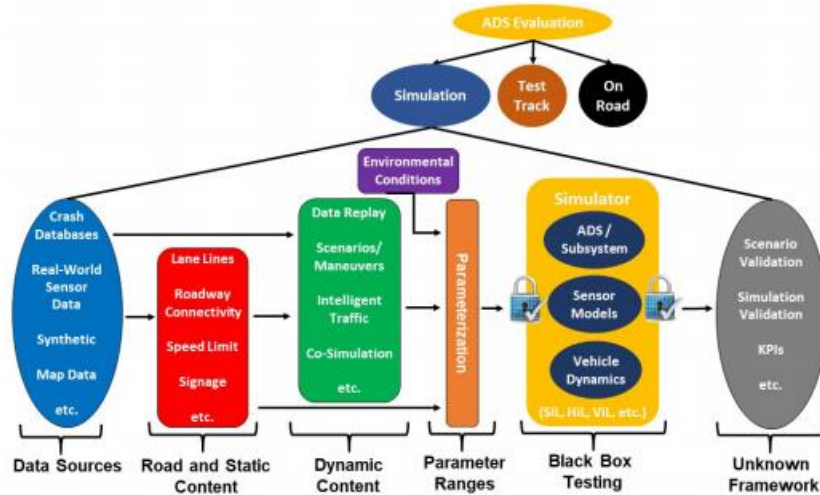


Figura 3.3.1. Proceso de prueba general de un sistema de navegación autónomo [39]

El modelado y simulación son herramientas utilizadas en conjunto para el análisis, diseño, adquisición y entrenamiento en la industria automotriz. Existen diferentes metodologías de simulación y pruebas que en conjunto hacen posible el desarrollo de los sistemas y subsistemas de un vehículo, una de las más utilizadas es la llamada *X-in-the-Loop* (XIL), dependiendo del nivel de desarrollo se utilizan las siguientes metodologías: *Model in the loop* (MIL), *Software in the loop* (SIL), *Hardware in the loop* (HIL) y *Vehicle in the loop* (VIL). En general un simulador puede hacer uso de estas metodologías, cada una de estas metodologías son utilizadas en diferentes etapas de desarrollo de los vehículos autónomos y en el caso particular de los sistemas ADAS, tal y como se observa en a Figura 3.3.2 [26][41].

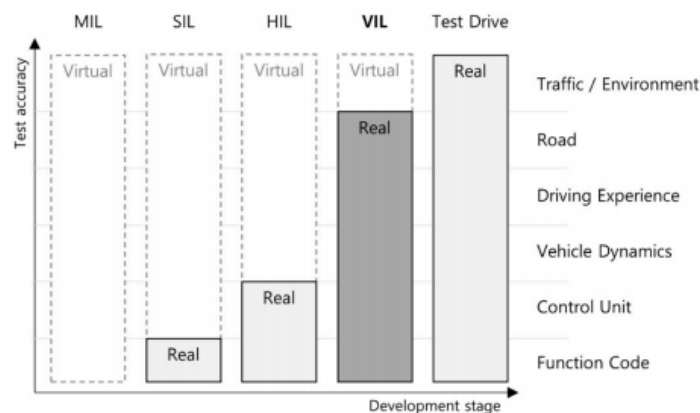


Figura 3.3.2. Comparación de las metodologías MIL, SIL, HIL, VIL y prueba de conducción real [41]

### 3.3.1 Model in the Loop-MIL

La metodología MIL es la primera etapa de desarrollo de un sistema, así pues, es la metodología aplicada a lo largo del diseño e implementación en este escrito, dicho desarrollo se detalla a lo largo del capítulo 5,6 y 7. El enfoque de la metodología MIL permite el desarrollo de

algoritmos sin la necesidad inmediata del uso de Hardware especializado. Por otro lado, tiene las siguientes características principales:

- MIL es un proceso que se asocia al paradigma de desarrollo basado en el modelado, también relacionado con una rama de la ingeniería llamada MBSE (Model-Based Systems Engineering).
- Requiere de la elaboración de una planta para el desarrollo del sistema/controlador en cuestión mediante el uso de diagramas como una representación del mundo real, se hace uso de herramientas como Simulink para tal propósito.
- El modelo propuesto se verifica mediante el controlador propuesto, por lo tanto, se comprueba que el controlador es capaz de satisfacer la planta del sistema bajo los diferentes posibles escenarios por analizar y comprobar.
- La finalidad de la metodología MIL, es utilizar su salida como herramienta de verificación durante las siguientes etapas de desarrollo SIL, VIL, etc. [26][42].

### ***3.3.2 Software in the Loop-MIL***

El segundo paso a considerar en una validación, es mediante la metodología SIL, esta consiste en la implementación del modelo desarrollado y su evaluación mediante hardware de propósito general, sus características principales son:

- Requiere de la implementación de un software muy cercano a la versión final.
- Describe una metodología de prueba en la cual el código ejecutable (puede ser código en C) se genera del modelo propuesto para el controlador en cuestión.
- Se reemplaza el modelo del controlador directamente con el código generado, posteriormente se realizan simulaciones con este y el modelo propuesto de la planta.
- Los resultados de esta prueba se cotejan con los resultados previos en la metodología MIL, es un punto de control de revisión para el desarrollo de sistemas y en caso de existir una discrepancia en ellos realizar alguna modificación.
- Técnica útil para corroborar el funcionamiento del software en desarrollo desde el comienzo de las etapas de diseño [26][42].

### ***3.3.3 Hardware in the Loop-HIL***

La metodología HIL es el siguiente paso en el desarrollo y validación de un sistema, sus principales características son:

- Involucra el hardware final que se usará en el diseño y es el proceso de validación del software final con sus respectivas entradas y salidas conectadas a un simulador.
- Provee una forma de simular, sensores, actuadores y componentes mecánicas de manera que las entradas y salidas de las unidades de control (ECUs) sean probadas.
- Hace uso de respuestas en tiempo real, estímulos eléctricos y posibles casos funcionales de uso.
- En complementación, la metodología SIL y HIL permiten a los diseñadores e ingenieros la evaluación en el desarrollo de los sistemas del vehículo antes de que se lleve a cabo un prototipado físico [26][42].

### ***3.3.4 Vehicle in the Loop-VIL***

Por lo general, las metodologías anteriores MIL-SIL-HIL son aceptables para el proceso de desarrollo de controladores convencionales, como pueden ser el sistema de ABS y el ESC por mencionar algunos; sin embargo, en el caso de los sistemas ADAS pueden llegar a requerir el uso de una metodología adicional, cuyo nombre es VIL. En el caso de esta metodología, su propósito principal recae en utilizar un ambiente virtual simulado probando los componentes de interés del sistema en desarrollo, sus características más destacadas son:

- Es la unión entre un entorno virtual y un vehículo de pruebas utilizado en el mundo real.
- La dinámica del vehículo se puede representar al mismo nivel que en el caso de uso real, lo cual significa una disminución en los costos al no tener que construir un entorno externo especializado para cada sistema.
- Se elimina el riesgo de una colisión durante pruebas reales y puede ser usado para el desarrollo de múltiples sistemas y pruebas como el desarrollo de un controlador de intersección, el desarrollo de un entorno para validar controladores de los sistemas ADAS mediante el uso de vehículos virtuales, etc. [26][41].

### 3.4 Plataformas de simulación

Si bien las metodologías anteriores son aplicables en muchos casos, es importante conocer cuáles son las plataformas existentes que facilitan su uso, dentro de las cuales podemos encontrar las siguientes: PaTAVTT, Simulink & Matlab, NI Monodrive con LabView, CarSim, CARLA y LGSVL [26].

#### 3.4.1 PaTAVTT

PaTAVTT (del inglés *platform for testing autonomous vehicle trajectory tracking*) es una plataforma de simulación y uso de la metodología HIL, consiste en una pista a escala que funciona con vehículos a escala, sistemas de posicionamiento e interfaces gráficas de usuario. Una de sus características destacables es su sistema de posicionamiento, ya que cuenta con un sistema interno de posicionamiento basado en la tecnología de banda ultra ancha (UWB), esta es comparada con respecto a la posición proporcionada por la tecnología GPS para un registro de posición mucho más certero. Adicionalmente, los modelos y parámetros, que son validados con el uso de Matlab o CarSim, pueden ser probados en un vehículo a escala y es posible transferirlos de esta plataforma al diseño de software de un vehículo real.

Consiste en una estructura conformada por un vehículo y pista escalados, un sistema de transmisión y un simulador de software. Tal y como se observa en la Figura 3.4.1.1, la plataforma tiene dimensiones rectangulares con una pista redondeada en los extremos en donde se mueven los vehículos que se pondrán a prueba, la comunicación y envío de información se lleva a cabo mediante el uso de tecnología inalámbrica de Wi-Fi.

La plataforma PaTAVTT puede ser utilizada para probar y verificar el funcionamiento de diferentes controladores tales como el control longitudinal y lateral del vehículo, funciones de mantenimiento y cambio de carril, evasión de obstáculos, control de cruceo adaptativo, entre otros [26][43].

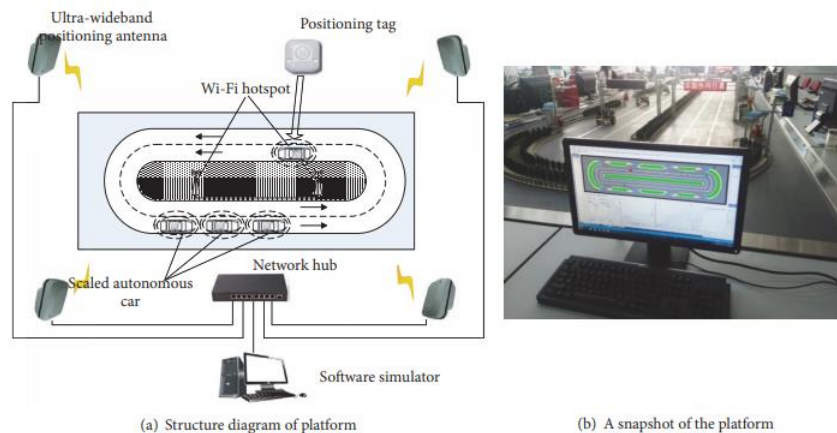


Figura 3.4.1.1. Estructura de la plataforma PaTAVTT [43]



### 3.4.2 Mathworks, Matlab & Simulink

Matlab es un ambiente de programación enfocado en el desarrollo de algoritmos, análisis de datos, visualización y computación numérica. Por otro lado, Simulink es un ambiente gráfico para simulación y diseño basado en el modelado de sistemas dinámicos e integrados de multi dominio, de igual forma es utilizado mayoritariamente en el desarrollo de controladores para el funcionamiento de los sistemas ADAS y puede ser integrado directamente en las unidades de control mediante la generación del algoritmo correspondiente [26][44].

Sus funciones principales son la obtención de información del comportamiento del mundo real, disminución del número de pruebas de los sistemas de un vehículo y validar el funcionamiento del software integrado, lo anterior se desglosa en las siguientes funciones y a su vez se puede observar en la Figura 3.4.2.1.

- **Acceso, visualización y etiquetado de datos.** Acceso a datos de conducción previamente registrados, mediante el uso de interfaces de Matlab para CAN y ROS.
- **Simulación de escenarios de conducción.** Utilización de entornos de simulación de cuboides y *Unreal Engine* para el desarrollo de los controladores en escenarios virtuales. Mediante el uso del primer entorno los vehículos se representan con gráficos simples y es utilizado para el desarrollo de sistemas de control, implementación de sensores y planificación de movimiento. El segundo, es utilizado para desarrollar algoritmos para casos prácticos de percepción.
- **Diseño de algoritmos de planificación y control.** Como su nombre lo indica permite el diseño de sistemas de control lateral y longitudinal de los vehículos autónomos que permiten el seguimiento de una trayectoria deseada.
- **Diseño de algoritmos de percepción.** Estos incluyen el uso de datos de cámara, de LiDAR y de radar. Se puede utilizar para aplicaciones de frenado automático, viraje, odometría y elaboración de mapas.
- **Despliegue de algoritmos.** Despliegue de algoritmos mediante la generación de código en C, C++, CUDA, Verilog y VHDL. Así mismo, estos se pueden utilizar para conectar sensores con la ECU y en arquitecturas de servicio como ROS o AUTOSAR.
- **Integración y pruebas.** Permite la integración de los sistemas de percepción, planificación y control mediante el uso de herramientas especializadas para capturar y gestionar requisitos; así como, ejecutar y automatizar casos de prueba en paralelo [26][44].

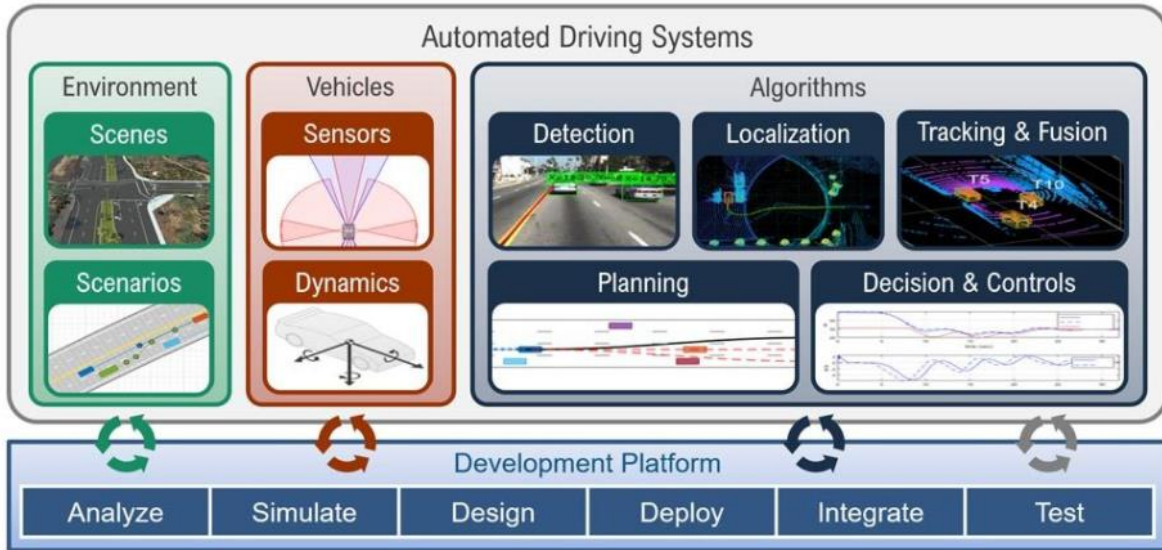


Figura 3.4.2.1. Estructura de la plataforma de diseño para sistemas de conducción autónoma por Mathworks [44]

### 3.4.3 National Instrument, Monodrive

MonoDrive es un simulador de vehículos autónomos fundado en 2016 y adquirido posteriormente en el año 2021 por National Instruments. Su propósito principal es la generación de escenarios realistas y de información sensorial para la validación de sistemas de percepción y navegación; de igual forma, busca lograr una baja latencia de información para realizar el control del vehículo; por otro lado, hace uso de una parametrización del modelado de la dinámica vehicular y de los sensores en tiempo real, en la Figura 3.4.3.A se puede observar su funcionamiento [45][46].

Se subdivide en las siguientes funciones principales:

- **Simulador:** es en donde se realizan pruebas de algoritmos de vehículos autónomos en niveles predeterminados. Este simulador es compatible mediante los protocolos de comunicación TCP/IP con cualquier aplicación que conforme la API (interfaz de programación de aplicaciones) de monoDrive; sin embargo, existe una mayor documentación de su uso con clientes como C++, Labview, y Python. El simulador ofrece cuatro modos de prueba, *closed loop* (lazo cerrado, activamente permite el control del sistema de sensores y una realimentación del vehículo), *replay* (ejecuta una secuencia fija con posiciones controladas sin realimentación), *HIL (hardware in the loop)* y *closed loop with fixed step* (permite el control de la diferencia del paso de la simulación sin tomar en cuenta el reloj de tiempo de simulación).
- **Editor de escenarios:** como tal es una extensión que permite el control y personalización de los escenarios de interés por el usuario, de esta manera permite maximizar el uso del

simulador y de los algoritmos probados en el vehículo. Puede generar comportamientos de conducción, parámetros de los vehículos y simulación de mapas personalizados.

- **Real a virtual:** es un hardware y software que provee una solución para el almacenamiento y recolección de información real proveniente de sensores como cámaras, LiDAR, y GNSS con el propósito de crear escenarios de alta fidelidad con respecto a dicha información, para ello hace uso del motor gráfico *Unreal Engine*. Funciona mediante el uso combinado de los sensores y algoritmos de *machine learning* que construyen un mallado en 3D y mediante un texturizado se reconstruye el mallado en el escenario deseado tal y como se observa en la Figura 3.4.3.B [46].

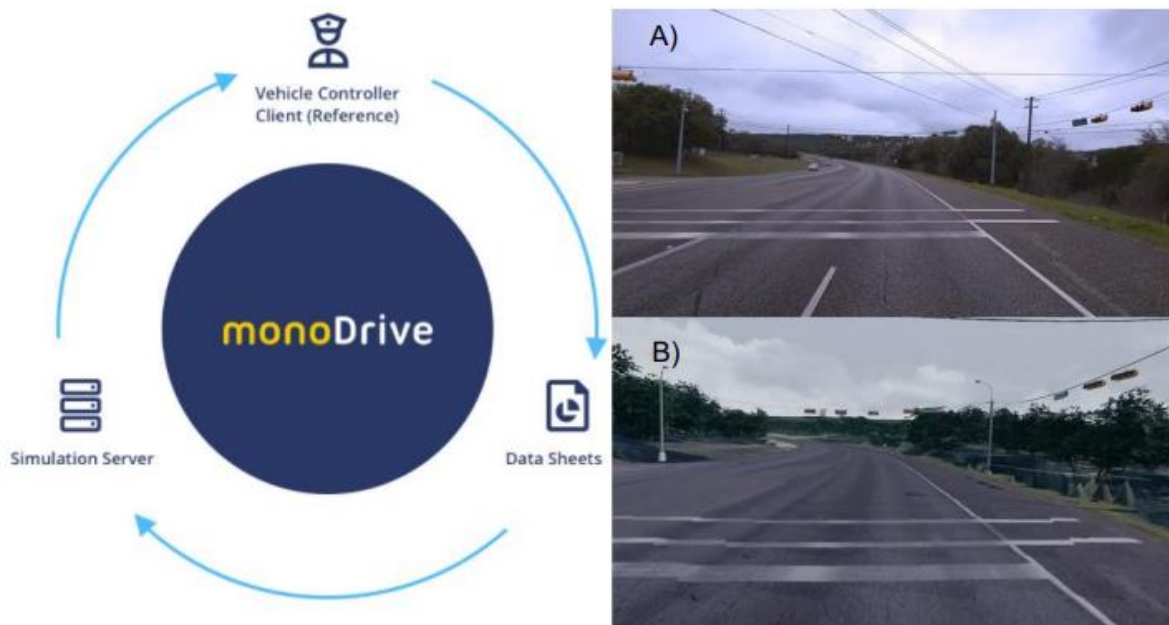


Figura 3.4.3. A) Funcionamiento de monoDrive. B) reconstrucción de escenario a partir de datos reales en un entorno virtual, imagen superior Real, imagen inferior virtual [46]

### 3.4.4 CarSim

CarSim es una herramienta de software utilizada ampliamente en la industria automotriz por grandes OEMs alrededor de todo el mundo, su propósito principal es la simulación del comportamiento dinámico de vehículos, así como de subcomponentes y controladores de seguridad probados en escenarios simulados de conducción. El tipo de vehículo en el que se especializa es en los de uso personal y camionetas de trabajo ligero, para ello hace uso de modelos dinámicos con los cuales logra reproducir los fenómenos físicos presentes en los vehículos, esto en respuesta a un controlador o acciones realizadas por un conductor.

Sus principales características y funciones se definen a continuación:

- Modelado vehicular de alta fidelidad:** los modelos matemáticos son representados a nivel de sistema, esto significa que los datos serán medidos o calculados y no dependerán de otras propiedades de los componentes, tales como su material o la complejidad de su estructura como la unión de los elementos de una suspensión. Dicho modelado contiene el suficiente nivel de información para asegurar la repetibilidad de los resultados e inclusive, estos pueden mejorar con el uso de información de referencia que sea relevante tales como: la cinemática de la suspensión, fuerza de tracción en los neumáticos seleccionados y condiciones ambientales.
- Definición paramétrica del vehículo:** los parámetros del vehículo pueden ser representados y obtenidos a partir de tablas, información publicada, herramientas de ingeniería, y pruebas industriales de los componentes de interés. Esto permite la modificación de los parámetros seleccionados en cualquier instancia del ciclo de diseño.
- Medición del desempeño del vehículo:** Modelos de conductor de lazo cerrado y abierto para la obtención óptima de los límites de capacidad del vehículo, se busca obtener una solución adecuada para mejorar la tarea de conducción. Su uso cumple con la certificación ISO y las regulaciones de control de estabilidad ECE.
- Herramientas de diseño estándar:** la simulación mecánica tiene interfaces muy bien acopladas a otros estándares de simulación y herramientas de diseño como Matlab/Simulink y NI/Labview, de igual forma se puede desarrollar tecnología mediante el uso de Visual Studio y de la propia API de CarSim integrada con Python como lenguaje de programación.
- Variedad de aplicaciones:** CarSim puede ser utilizado con las metodologías MIL, SIL, HIL, y DIL; de igual forma, es utilizado en el desarrollo de sistemas como: frenado ABS, ESC, ACC, suspensiones activas, conducción autónoma, tecnología ADAS, estudios de eficiencia de combustible entre otras más [47].

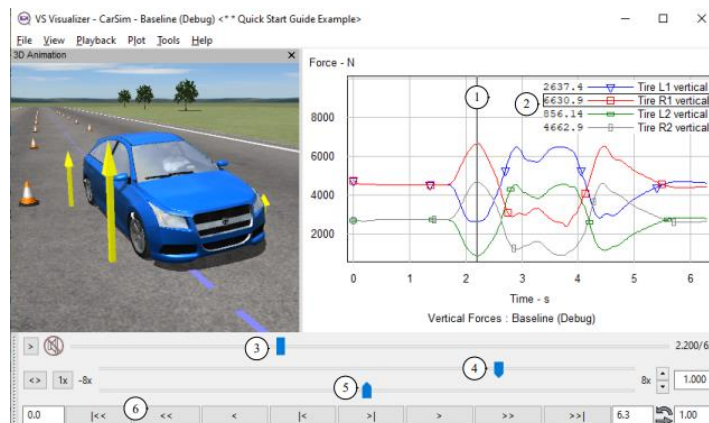


Figura 3.4.4. Visualización en CarSim y herramientas de graficación [47]

### 3.4.5 Carla

Es un simulador de código abierto dedicado a la investigación, desarrollo, entrenamiento y validación de sistemas de navegación autónoma, en este caso vehículos autónomos. Su principio básico funcionamiento consiste en una estructura de cliente-servidor; esto implica que el servidor es el encargado de todo aquello que se relaciona con la simulación, como el procesamiento de los sensores, de la física, así como la actualización de los actores y el escenario representado. Por otra parte, el cliente controla las condiciones del escenario y la lógica de los actores dentro del mismo mediante la API integrada en el simulador CARLA (en C++ o Python), en la Figura 3.4.5 se puede observar dicha estructura y la interfaz gráfica del simulador [48][49]. Dentro de las características más destacadas se encuentran:

- **Gestor de tránsito:** es un sistema que toma control de todos los vehículos en el escenario incluyendo el vehículo de estudio. Su propósito principal es el de recrear entornos urbanos en los que se reproduzcan comportamientos realistas de los mismos.
- **Sensores:** tal como sucede en cualquier simulador, son parte fundamental para la obtención de información del entorno. En este caso, el simulador CARLA cataloga los sensores como actores independientes de la simulación, lo que facilita el almacenamiento de la información capturada por los mismos. Se pueden encontrar entre ellos cámaras, radares, lidar, etc.
- **Grabador:** esta función permite la recreación de una simulación paso por paso para cualquiera de los actores en el escenario, por lo cual, resulta una herramienta útil para el monitoreo y rastreo en algún momento dado del proceso.
- **Implementación con ROS-bridge y Autoware:** el simulador CARLA tiene la facilidad de integrarse en conjunto con otros entornos de aprendizaje, tales como ROS y Autoware.
- **Generación de mapas:** permite la creación de mapas personalizados por el usuario, de igual manera, se pueden controlar las condiciones climáticas y tiene disponible una gran variedad de actores para utilizar.
- **Escenarios predeterminados:** con el propósito de facilitar el aprendizaje de los vehículos, CARLA contiene rutas predeterminadas en los escenarios que describen diferentes situaciones en las cuales se puede iterar continuamente [48][49].

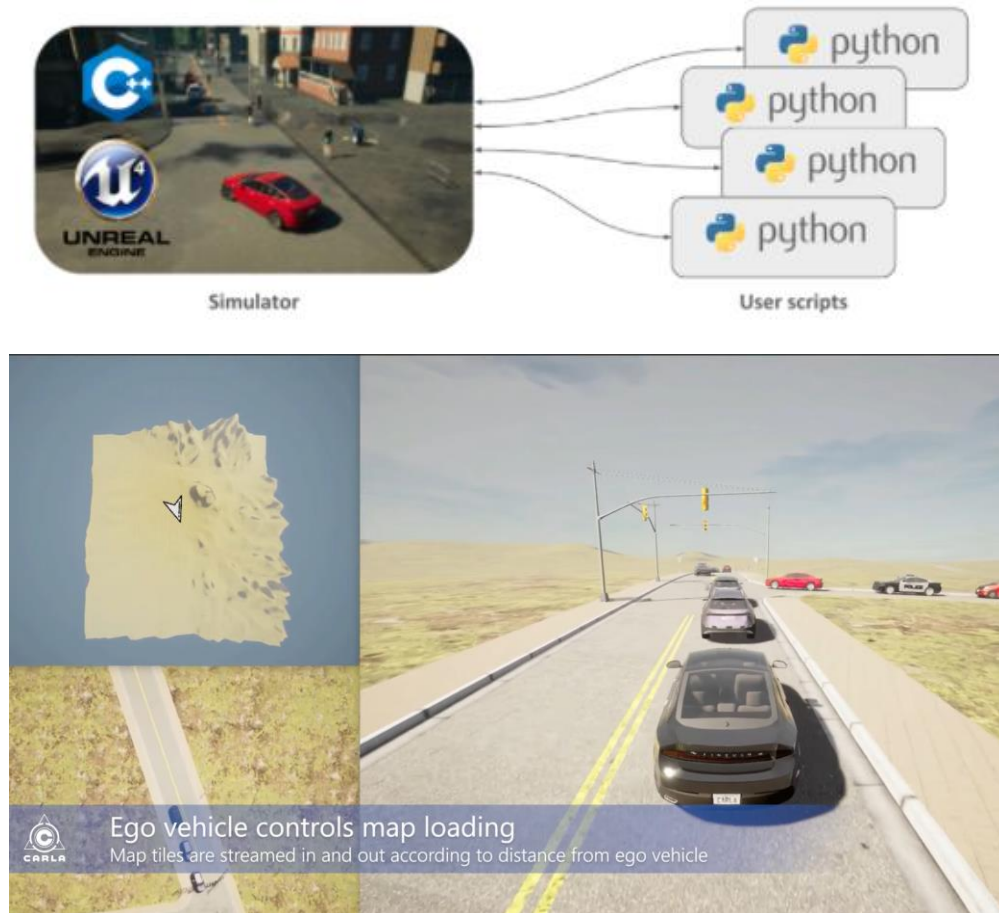


Figura 3.4.5. Superior: estructura de funcionamiento de CARLA. Inferior: Ejemplo de simulación del simulador CARLA mostrando la carga del mapa utilizado. [49]

### 3.4.6 SVL

SVL simulator, es un simulador de código abierto desarrollado por *LG Electronics America R&D Lab* en el año 2018, con el propósito de ayudar a los desarrolladores a construir vehículos y robots autónomos mediante el uso de simulaciones en 3D, actualmente se ha vuelto un producto comercial ayudando a muchas OEM's automotrices, compañías sobre robótica e inclusive universidades a lo largo del mundo a llevar a cabo sus desarrollos.

El simulador consta de un software dedicado a la simulación, herramientas de software y plugins diseñados para casos particulares de uso. Con SVL simulator se pueden realizar pruebas de integración, pruebas de algoritmos (localización, percepción y planeación de rutas, tal y como se observa en la Figura 3.4.6) y verificación de sistemas, además es capaz de proveer salida de datos de sensores como cámaras, LiDAR, radar, GPS e IMUs, entre otros.

Algunas de sus características principales son:

- Creación e implementación de escenarios de tránsito complejos basados en información obtenida del mundo real.
- Mejoramiento y depuración del módulo de localización mediante pruebas en entornos con modelos virtuales apegados al mundo real.
- Pruebas del módulo de planeación en aislamiento con la detección de información conocida previamente establecida de manera virtual en el simulador.
- Prueba de vehículos autónomos mediante el uso de las metodologías SIL y HIL.
- Modificación de mapas, condiciones ambientales, luz del entorno y desarrollo de escenarios mediante el uso de código en C++ y Python [50][51].

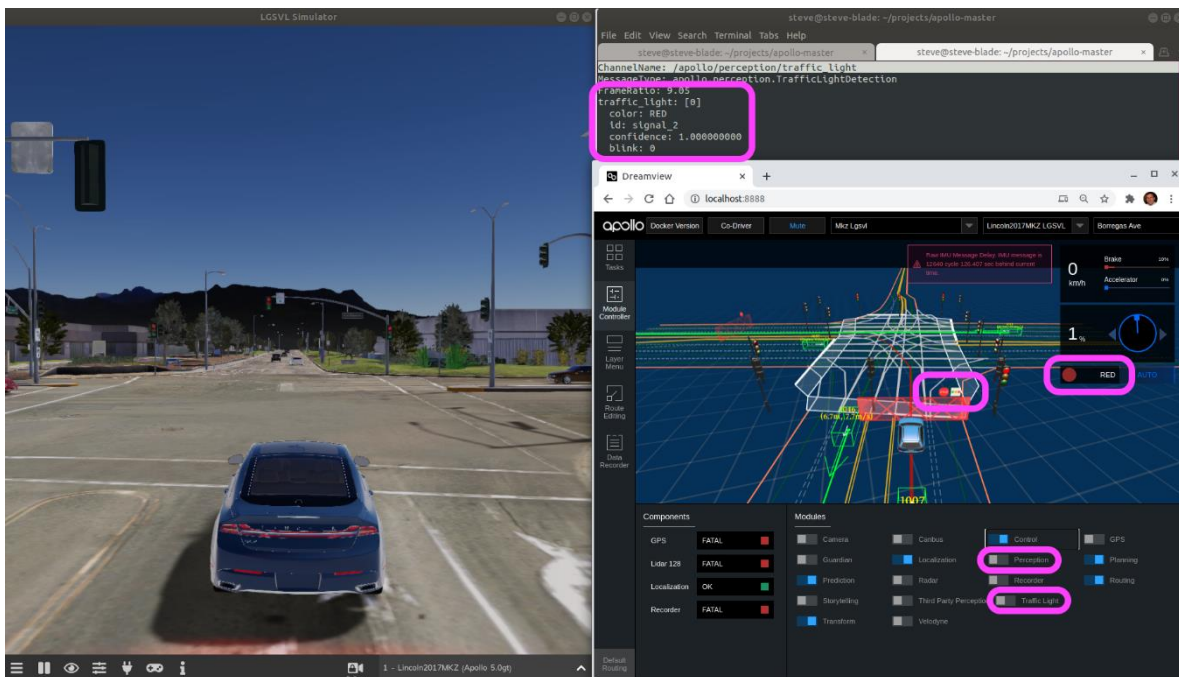


Figura 3.4.6. Interfaz gráfica del simulador SVL, sistema de percepción [50]

### 3.5 Selección de plataforma de simulación

A lo largo del capítulo se han encontrado una gran variedad de simuladores y plataformas que complementan el desarrollo de sistemas de conducción autónomos; sin embargo, es necesario realizar una pequeña comparación para elegir la herramienta que más se acople a los objetivos de este trabajo de investigación. Puesto que uno de ellos es el desarrollo y comparación de un mismo controlador con diferentes metodologías o leyes de control, se considera que para su desarrollo y aplicación se encuentra alrededor de la metodología MIL. Por tal razón este será un criterio de vital importancia a considerar en la elección, por otro lado, también vale la pena

considerar su compatibilidad con otros sistemas y el tipo de licencia de cada uno de ellas. Para tal propósito se realizó una pequeña comparación de las plataformas descritas anteriormente tal y como se registra en la Tabla 3.5.

Simulador	Tipo de Licencia	ISO 26262	MIL	XIL	
				SIL	HIL
PaTAVTT	Libre/código abierto	I	-	-	X
Simulink & Matlab	Comercial	X	X	X	X
MonoDrive	Comercial	X	X	X	X
CarSim	Comercial	X	X	X	X
Carla	Libre/ código abierto	I	I	I	I
LGSVL	Libre/código abierto	I	-	X	X

Legenda: "x" - Sí, "I"-indeterminado, "-" - No

Tabla 3.5. Resumen comparativo de algunas características de las plataformas de simulación adaptado de [26].

Lo anterior resulta en la necesidad de contar con una plataforma de simulación que ofrezca la posibilidad de desarrollar e implementar dichos modelos; así como, una manera de probarlos en el mundo real, por lo tanto, se deben considerar las etapas de prueba posteriores al modelado, es decir, un proceso de desarrollo completo mediante el uso de las metodologías MIL, SIL y HIL en conjunto. De la Tabla 1, solo tres plataformas de simulación permiten el uso de estas metodologías, los cuales son Simulink & Matlab, MonoDrive y CarSim, si bien el resto no contempla como tal cada una de las metodologías de prueba, son herramientas relevantes utilizadas en el desarrollo de los vehículos autónomos y que comparten un gran número de características entre sí. De las tres plataformas de interés, cabe destacar que todas ellas cumplen con las regulaciones de la norma ISO 26262, que en un muy alto nivel se refiere a un estándar internacional que regula las funciones específicas necesarias para la seguridad funcional que deben cumplir los sistemas, procesos, métodos y herramientas durante el proceso de desarrollo de los vehículos. [52] Adicionalmente, estas plataformas son compatibles en algunos casos entre sí, dependiendo de los clientes de programación de los cuales hagan uso, por lo que algunos aspectos extras tomados en consideración para esclarecer la elección fueron: familiaridad y accesibilidad. Siendo que todas las opciones requieren de una curva de aprendizaje similar para adaptarse a sus funciones y herramientas, este aspecto se vuelve secundario al problema de accesibilidad, el cual se resuelve gracias al acceso que nos otorga la UNAM para hacer uso de ciertos programas, entre ellos Matlab & Simulink. Aunado a lo anterior, Matlab tiene un cuadro de herramientas específico para el desarrollo del diseño de los sistemas ADAS, esto implica que cuenta con códigos y herramientas de utilidad específicos para el desarrollo del sistema ACC [26].

Por lo anterior, se definió como herramienta de trabajo a la plataforma de simulación de Matlab & Simulink, ya que posee las características y requerimientos necesarios para hacer posible el desarrollo de nuestro sistema ACC de la manera más óptima posible.



### **3.6 Conclusión**

Durante este capítulo ha sido posible entender sobre el funcionamiento de las plataformas de simulación, su papel y durante que etapas en el desarrollo/diseño de un vehículo pueden ser utilizadas como herramientas complementarias. Muchas de ellas comparten características de su funcionamiento; sin embargo, la mejor opción siempre se puede determinar identificando el cumplimiento de las necesidades planteadas para el diseño de nuestros sistemas, esto quiere decir que todas ellas son de utilidad en la medida que uno lo plantee y al nivel de trabajo que se quieran considerar. Adicionalmente, en el capítulo se presentaron algunas plataformas que son utilizadas con frecuencia; sin embargo, existe una variedad mucho mayor de las aquí propuestas, cada una de ellas adaptadas a diferentes necesidades.



# ***4. Diseño del Control de Crucero Adaptativo (ACC)***

Hasta ahora se ha revisado el estado del arte que gira en torno a los vehículos autónomos, los sistemas ADAS que lo conforman y las metodologías de pruebas utilizadas en la industria para su validación mediante plataformas de simulación, por lo que en esta sección se detallan los alcances de la propuesta de diseño y cuál será el proceso para conseguir nuestros objetivos primarios descritos en el Capítulo 1.

## ***4.1 Definición del problema***

La creciente congestión vehicular, hace que los conductores de las grandes urbes sean más propensos a tener un accidente, siendo el error humano uno de los factores principales causantes de esto. Por lo tanto, resulta de vital importancia el mejoramiento de los sistemas de control en los vehículos, lo cual implica una mejora en la seguridad de los mismos.

Uno de los principales sistemas de control destinados a mejorar la seguridad vehicular y el confort del conductor al manejar un vehículo es el control de crucero adaptativo (ACC). Su tarea central consiste en el control de la dinámica longitudinal del vehículo.

En este sentido se pretende estudiar y evaluar el desempeño de un ACC virtualmente, para ello este estudio involucra las siguientes etapas de desarrollo:

- **Modelado**

Se presentan conocimientos sobre el modelado de los controladores propuestos para el control del sistema ACC; de igual manera se hace uso de los conceptos sobre dinámica vehicular, con la finalidad de implementarlos mediante Simulink y Matlab.

- **Banco de Pruebas Virtual**

En este apartado se describe con detalle el uso que se le dará a la plataforma de simulación, así como la adición de los modelos anteriores de manera funcional en el sistema seleccionado.

- **Metodología y Resultados**

En esta etapa de desarrollo se realiza el diseño y evaluación de los controladores modelados anteriormente, se busca mantener la mayor cantidad de variables bajo control para realizar una comparación del desempeño bajo una selección de parámetros para cada escenario en particular.

## 4.2 Propuesta de Solución

En concordancia con lo anterior, se desarrollarán las siguientes cuatro estrategias de control para el desarrollo del ACC:

- Clásico control proporcional-integral-derivativo (PID)
- Espacio de tiempo constante (*Constant Time Gap* – CTG)
- Control deslizante (*Sliding Mode Control* – SMC)
- Control predictivo por modelo (*Model Predictive Control* – MPC)

Lo anterior creará la oportunidad de comparar el desempeño de cada estrategia en seis escenarios destinados a determinar el comportamiento del controlador en el vehículo ante situaciones particulares. Dichos escenarios se describen con mayor detalle en la sección de Metodología y Resultados. Véase Figura 4.2.

Mediante el uso de la plataforma de simulación también será posible visualizar gráficamente el comportamiento del vehículo con cada controlador, esto es posible gracias a la interfaz que presenta Matlab con el motor gráfico *Unreal Engine*.

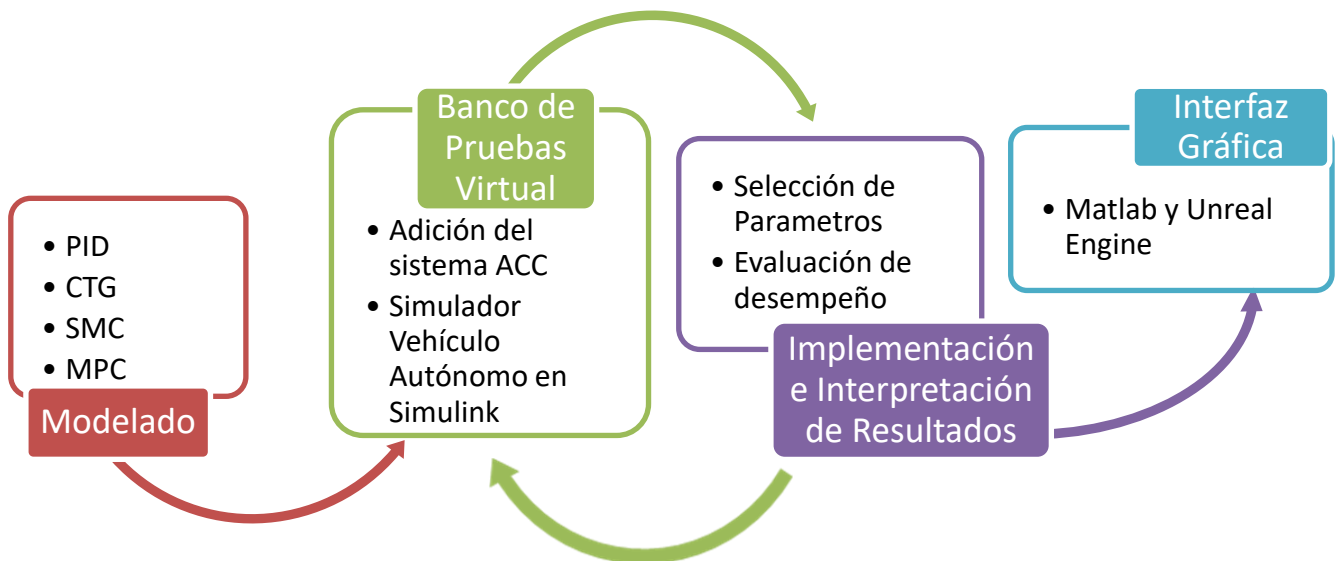


Figura 4.2. Desarrollo de la solución propuesta

Debido a que todas las iteraciones se realizarán de forma virtual, el alcance de este estudio está delimitado por las pruebas realizadas y el análisis de la información obtenida de las mismas.

## 4.3 Conclusión

En este capítulo se determinó la propuesta de solución para la problemática abordada, se logró delimitar el alcance de la misma, así como las herramientas y procedimientos a utilizar para su desarrollo.



## 5. Modelado del Sistema

Para poder resolver la problemática abordada en el capítulo anterior es necesario comprender nuestro sistema, para ello en se deben entender conceptos relevantes sobre la dinámica vehicular, la estructura del controlador de cruceo adaptativo (ACC) y las diferentes estrategias de control y su principio de funcionamiento para lograr el objetivo de control planteado.

### 5.1 Dinámica vehicular longitudinal

Si consideramos un vehículo moviéndose longitudinalmente en una pendiente y las fuerzas actuando en él, existen fuerzas de arrastre, de resistencia a la rodadura, así como la misma gravedad actuando en el vehículo, tal y como se observa en la Figura 5.1.

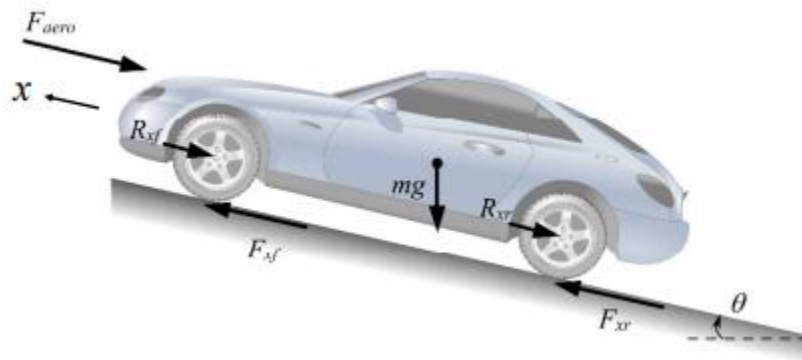


Figura 5.1. Diagrama de fuerzas longitudinales actuando sobre un vehículo en movimiento. Imagen extraída de [17].

Realizando un balance de fuerzas en el eje x se tiene que:

$$m\ddot{x} = F_{xf} + F_{xr} - F_{aero} - R_{xf} - R_{xr} - mg\sin(\theta)$$

Donde:

- $F_{xf}$  es la fuerza longitudinal de los neumáticos delanteros
- $F_{xr}$  es la fuerza longitudinal de los neumáticos traseros
- $F_{aero}$  es la fuerza aerodinámica equivalente a la fuerza de arrastre del vehículo
- $R_{xf}$  es la fuerza de resistencia a la rodadura de los neumáticos delanteros

$R_{xr}$	es la fuerza de resistencia a la rodadura de los neumáticos traseros
$m$	es la masa el vehículo
$g$	es la aceleración de la gravedad
$\theta$	es el ángulo de inclinación del camino por el que el vehículo se está desplazando.

El modelo plasmado en esta sección nos ayudará a determinar la dinámica vehicular lateral y longitudinal del vehículo bajo ciertas consideraciones que se detallaran posteriormente; sin embargo, es relevante mencionar que entre ellas se considera que la fuerza aerodinámica y la resistencia a la rodadura son nulas y por tal motivo no se describirán a detalle dichas fuerzas. No obstante, una de las fuerzas que si debe ser considerada es la fuerza longitudinal de los neumáticos. [17]

### **5.1.2 Fuerza aerodinámica**

La fuerza aerodinámica en el eje longitudinal está representada por la fuerza de arrastre, la cual es la fuerza que actúa en oposición al movimiento del vehículo avanzando a través del aire, dicha fuerza está dada por la siguiente expresión [53]:

$$F_{aero} = \frac{1}{2} \rho C_d A_F V_x^2$$

Donde:

$\rho$	es la densidad del aire
$C_d$	es el coeficiente de arrastre aerodinámico
$A_F$	es el área frontal del vehículo
$V_x$	es la velocidad longitudinal del vehículo

### **5.1.3 Resistencia a la Rodadura**

La resistencia a la rodadura se presenta en el momento que el neumático rueda sobre una superficie en donde existe una deformación de ambos elementos; sin embargo, la deformación de la superficie puede ser despreciada comparativamente hablando con la que sufre el neumático. Esta deformación se debe principalmente a la carga normal, la pérdida de energía que sufren los neumáticos durante este proceso se conoce como resistencia a la rodadura y esta actúa en oposición al movimiento del vehículo.

Por lo general, la resistencia a la rodadura es prácticamente proporcional a la fuerza normal en cada neumático y se representa de la siguiente manera [17]:

$$R_{xf} + R_{xr} = f(F_{zf} + F_{zr})$$

Donde:

$F_{zf}$  es la fuerza normal en los neumáticos delanteros

$F_{zr}$  es la fuerza normal en los neumáticos traseros

$f$  es el coeficiente de resistencia a la rodadura

### 5.1.4 Fuerza longitudinal del neumático

Las fuerzas longitudinales son aquellas que actúan directamente en los neumáticos a causa de la fricción causada con la superficie de contacto, denominadas como  $F_{xf}$  y  $F_{xr}$ .

Para entender de donde proviene la fuerza longitudinal se debe entender que un neumático es un elemento que se deforma debido a la carga vertical que se ejerce sobre él. Además, el área de contacto entre el neumático y el suelo se conoce como área de contacto, en la Figura 5.1.4.1 se pueden observar las fuerzas que actúan en el neumático.

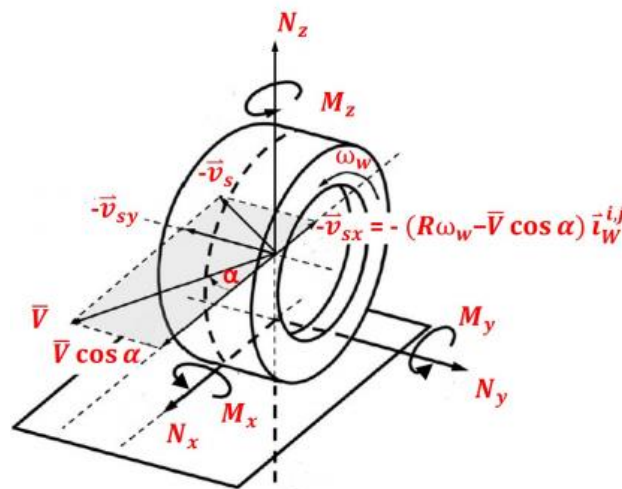


Figura 5.1.4.1. Fuerzas y momentos que actúan en el neumático. [Notes VDC]

Donde  $N_x, N_y$  y  $N_z$  son las fuerzas que actúan a lo largo de los ejes X, Y, Z en el área de contacto, de manera similar  $M_x, M_y$  y  $M_z$ , son los momentos generados en este punto de contacto [17].



## Ángulo de deslizamiento

Para continuar la descripción de los elementos en la Figura 5.1.4.1 es necesario comprender el ángulo de deslizamiento. Este se define como el ángulo entre la dirección del vector de velocidad del centro del neumático (eje B) y el eje longitudinal fijado en el centro del mismo (eje W), tal y como se observa en la Figura 5.1.4.2 [17].

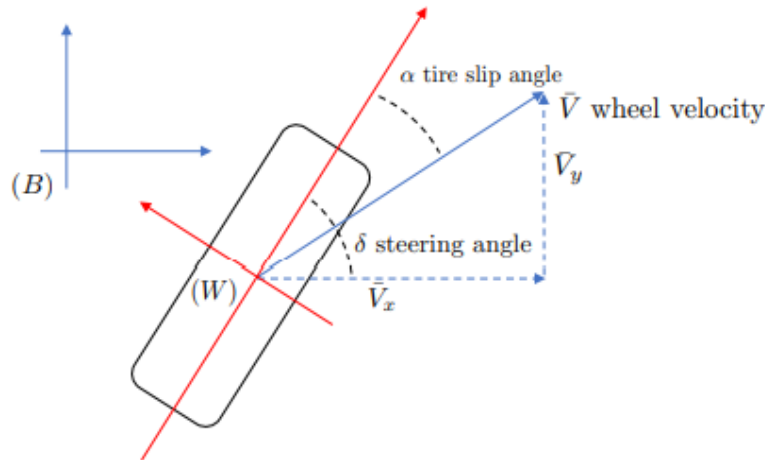


Figura 5.1.4.2. Ángulo de deslizamiento en el neumático

El ángulo de deslizamiento se calcula de la siguiente manera:

$$\alpha = \delta - \tan^{-1} \frac{\bar{V}_y}{\bar{V}_x}$$

Donde:

$B$  es el eje coordinado con respecto al cuerpo del vehículo

$W$  es el eje coordinado con respecto al centro del neumático

$\alpha$  es el ángulo de deslizamiento

$\delta$  es el ángulo de giro de la dirección

$\bar{V}_y$  es la componente de la velocidad del neumático en el eje y con respecto al eje coordinado B.

$\bar{V}_x$  es la componente de la velocidad del neumático en el eje x con respecto al eje coordinado B.

## Relación de deslizamiento

La relación de deslizamiento se puede definir como el cociente de la diferencia de velocidad que existe entre el punto de contacto del neumático con la superficie de interés

(deslizamiento longitudinal) y la velocidad longitudinal o la velocidad de rotación; lo anterior depende si es en una situación de frenado o aceleración, tal y como se observa a continuación [17]:

$$\sigma = \frac{R \omega_w - \bar{V} \cos(\alpha)}{\bar{V} \cos(\alpha)} = \frac{\bar{v}_{sx}}{\bar{V}_x} \quad \text{durante frenado}$$

$$\sigma = \frac{R \omega_w - \bar{V} \cos(\alpha)}{R \omega_w} = \frac{\bar{v}_{sx}}{R \omega_w} \quad \text{durante aceleración}$$

En la Figura 5.1.4.3 se representa la relación de deslizamiento.

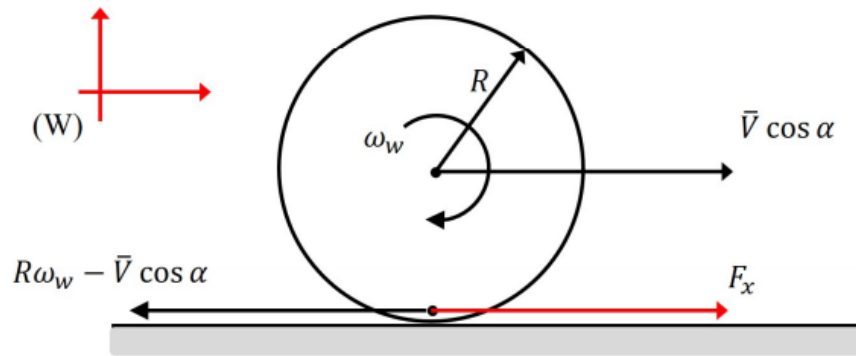


Figura 5.1.4.3. Representación de la relación de deslizamiento

Donde:

- $R$  es el radio efectivo del neumático
- $\omega_w$  es la velocidad de rotación del neumático
- $\bar{v}_{sx}$  es la velocidad de deslizamiento en dirección longitudinal

### Modelo Lineal Lateral y longitudinal

Una vez que se han determinado tanto el ángulo de deslizamiento como la relación de deslizamiento, las fuerzas lateral y longitudinal del neumático depende exclusivamente de los siguientes elementos:

$$F_y = f_y(\alpha, \sigma, \mu, F_z)$$

$$F_x = f_x(\alpha, \sigma, \mu, F_z)$$

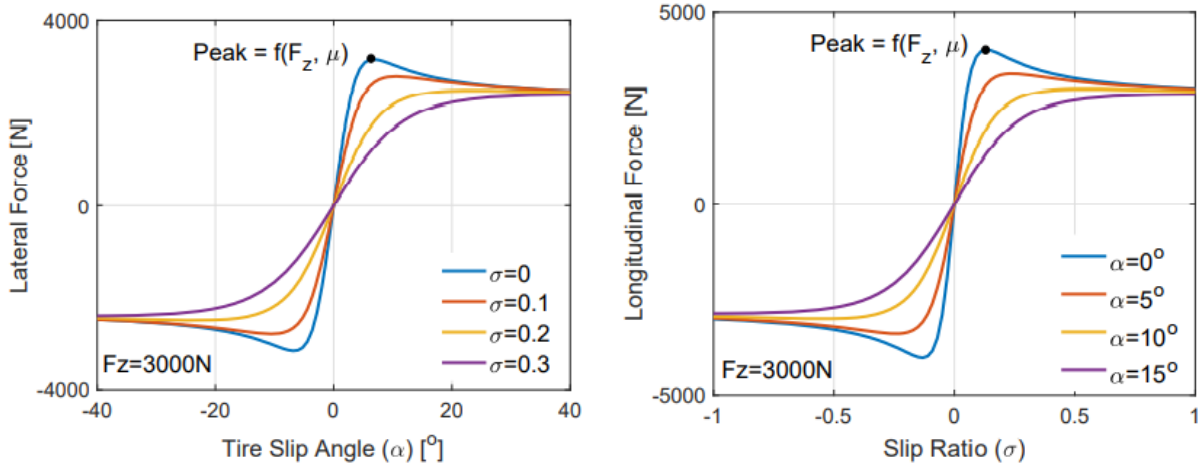


Figura 5.1.4.4. Lado izquierdo: fuerza lateral como función del ángulo de deslizamiento; lado derecho: fuerza longitudinal como función de la relación de deslizamiento.

Donde:

$\mu$  es el coeficiente de fricción de la superficie en contacto con el neumático

No obstante, cuando el neumático no opera cerca de la zona de saturación (posterior al pico) y el ángulo de deslizamiento, junto con la relación de deslizamiento son pequeños se puede aproximar el modelo de la fuerza lateral y longitudinal como se sigue a continuación con un modelo lineal [17]:

Ángulo de deslizamiento pequeño  $F_y = C_\alpha \alpha$

Relación de deslizamiento pequeño  $F_x = C_\sigma \sigma$

Donde:

$C_\alpha$  coeficiente de rigidez lateral

$C_\sigma$  coeficiente de rigidez longitudinal

## 5.2 Modelo dinámico de dos ruedas (Bicycle Model)

En este estudio se utiliza un modelo de dos ruedas (bicicleta del inglés bicycle) con 3 grados de libertad, considerando el movimiento lateral, longitudinal y el ángulo de giro del vehículo (ángulo de yaw). Para poder desarrollar este modelo se deben tener ciertas consideraciones [54]:

- Se asume que los neumáticos delanteros y traseros están unidos en una sola y por lo tanto se tiene un modelo que considera un ancho de rodadura cero (*track width*).
- Se desprecia el efecto de la fuerza de arrastre y la resistencia a la rodadura.
- Se considera que no existe pitch y roll.
- No hay transferencia de carga.
- El movimiento del vehículo está restringido al plano X-Y (Posición global del vehículo)

- Ángulo de giro de la dirección pequeño, únicamente giro delantero.
- Variación lenta de la velocidad longitudinal en el tiempo
- Solo fuerzas laterales en los neumáticos, generadas por un modelo lineal
- El control de la aceleración longitudinal se lleva a cabo desde el eje coordinado del vehículo (B)

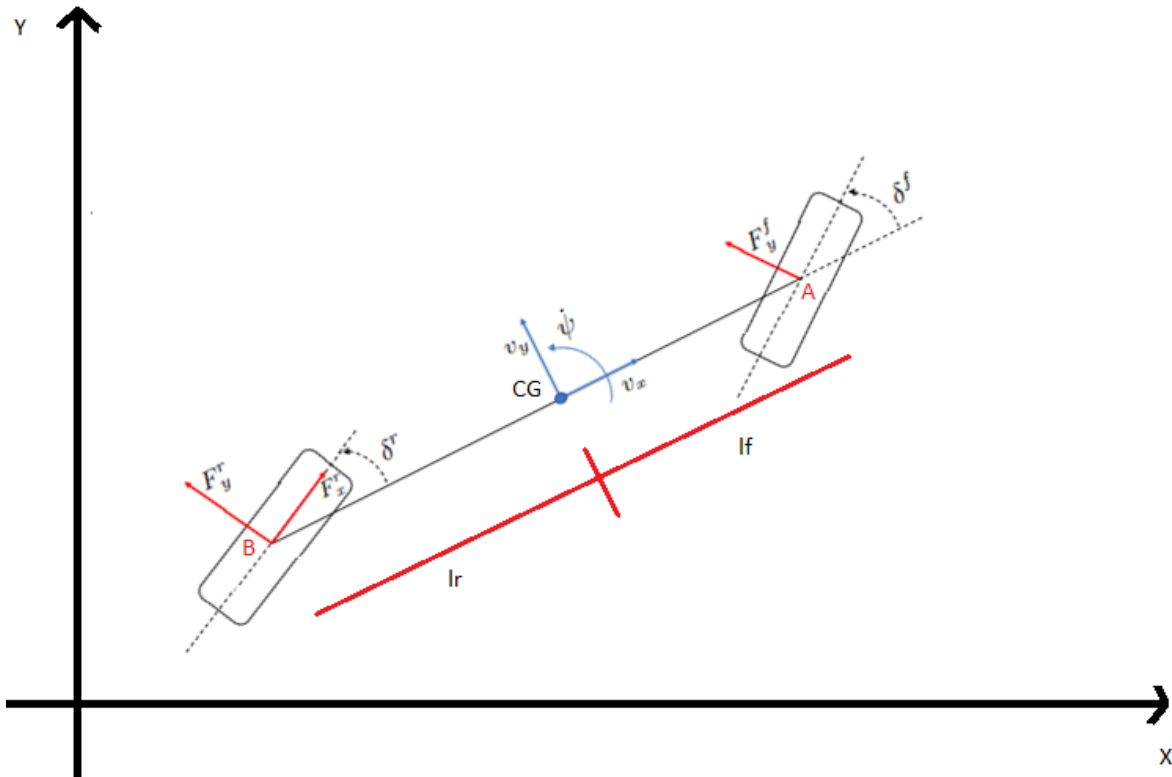


Figura 5.2.1. Dinámica lateral del vehículo con respecto a la posición global del mismo, plano X-Y

Utilizando la segunda ley de Newton se obtiene que la suma de fuerzas con respecto al eje y del centro de gravedad es:

$$ma_y = F_y^r + F_y^f \quad (1)$$

En donde:

$m$  es la masa del vehículo

$a_y$  es la aceleración inercial del vehículo en el centro de gravedad

$F_y^r$  y  $F_y^f$  son las fuerzas laterales de los neumáticos delanteros y traseros respectivamente

La aceleración  $a_y$  se calcula a partir de dos términos, la aceleración en el eje y, así como a aceleración centrípeta. Por lo cual:

$$a_y = \dot{v}_y + v_x \dot{\psi} \quad (2)$$

Donde:

$v_y$  y  $v_x$  son la velocidad lateral y longitudinal del vehículo respectivamente

$\dot{\psi}$  es la tasa de cambio de la posición con respecto al ángulo de yaw del vehículo, o lo que es equivalente a la velocidad angular  $\omega_z$

Si sustituimos la ecuación (2) en (1) se obtiene que:

$$m(\dot{v}_y + v_x \dot{\psi}) = F_y^r + F_y^f \quad \text{ecuación (3)}$$

Obteniendo la suma de momento sobre el eje z se tiene la siguiente ecuación para la dinámica con respecto al ángulo de giro yaw:

$$I_z \ddot{\psi} = l_f F_y^f - l_r F_y^r \quad (4)$$

Donde:

$I_z$  es el momento de inercia sobre el eje z

$l_f$  y  $l_r$  la distancia del CG al eje delantero y trasero respectivamente

Recordando el modelo lineal propuesto en la sección 5.1.4, la fuerza longitudinal es proporcional con respecto al ángulo de deslizamiento. De esta afirmación la fuerza lateral se aproxima de la siguiente manera para ambos ejes delantero y trasero, en este caso hemos asumido una dirección negativa:

$$F_y^r = -2C_\alpha^r \alpha_r \quad \text{eje trasero} \quad (5)$$

$$F_y^f = -2C_\alpha^f \alpha_f \quad \text{eje delantero} \quad (6)$$

De lo anterior se puede notar que se requiere obtener el ángulo de deslizamiento delantero y trasero, para tal propósito se requiere analizar los puntos de interés A y B de forma separada. Mediante el uso de geometría podemos descomponer la velocidad en estos puntos en sus componentes x,y, tal y como se muestra en la Figura 5.2.2.

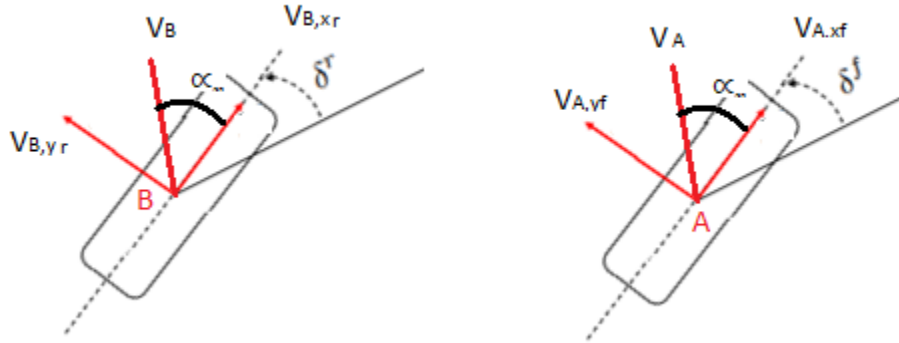


Figura 5.2.2. Lado izquierdo: descomposición de la velocidad en el neumático trasero (B); lado derecho: descomposición de la velocidad en el neumático delantero (A).

Si analizamos la velocidad en el punto B y A, se tiene que:

$$\alpha_r \approx \tan \frac{V_{B,yr}}{V_{B,xr}} \approx \frac{V_{B,yr}}{V_{B,xr}} \quad ; \quad \alpha_f \approx \tan \frac{V_{A,yf}}{V_{A,xf}} \approx \frac{V_{A,yf}}{V_{A,xf}} \quad (7)$$

Se sabe que en que la velocidad en el CG tiene dos componentes,  $V_x$  y  $V_y$  respectivamente. Haciendo uso de las leyes de la cinemática en un cuerpo rígido, dicha velocidad se transforma de la siguiente manera y tiene los siguientes valores en el origen de los ejes coordenados de cada neumático (punto B y A) tal y como se sigue a continuación:

$$V_{B,x} = V_x \quad ; \quad V_{B,y} = V_y \quad (8)$$

$$V_{A,x} = V_x \quad ; \quad V_{A,y} = V_y + \omega l_r \quad (9)$$

Ahora bien, para determinar las componentes de la velocidad en los ejes coordenados (xr,yr), (xf,yf) y recordando que  $\delta_f \neq 0$  y  $\delta_r = 0$ ; es claro ver que en el punto B, al tener una rotación cero en el eje trasero, la velocidad del eje (xr,yr) será igual a las componentes de la velocidad en el punto B. Es decir:

$$V_{B,xr} = V_{B,x} \quad ; \quad V_{B,yr} = V_{B,y} \quad (10)$$

Por el contrario en el punto A, el ángulo de rotación de giro hace que se deban proyectar las componentes de la velocidad en (xf,yf) al eje coordenado del CG (x,y), para ello se lleva a cabo la siguiente transformación de coordenadas:

Para un  $\delta$  es pequeño

$$V_{A,xf} = V_{A,x} \cos \delta + V_{A,y} \sin \delta = V_{A,x} + V_{A,y} \delta \quad (11)$$

$$V_{A,yf} = V_{A,y} \cos \delta - V_{A,x} \sin \delta = V_{A,y} - V_{A,x} \delta \quad (12)$$

Sustituyendo las ecuaciones (7), (8), (9), (10), (11) y (12) en (5) y (6):

$$F_y^r = -2C_\alpha^r \alpha_r = -2C_\alpha^r \left( \frac{V_y - \omega l_r}{V_x} \right) \quad (13)$$

Asumiendo:  $V_x \gg V_y - \omega l_r$  se tiene que:

$$F_y^f = -2C_\alpha^f \alpha_f = -2C_\alpha^f \left( \frac{-V_x \delta + V_y - \omega l_r}{V_x} \right) = 2C_\alpha^f \delta - 2C_\alpha^f \left( \frac{V_y - \omega l_r}{V_x} \right) \quad (14)$$

De lo manera anterior ha sido posible determinar la fuerza lateral del eje delantero y trasero; sin embargo, se requiere representar estas fuerzas y ecuaciones en una representación de espacio de estados [54].

### **Representación en espacio de estados**

Para llevar a cabo la representación en espacio de estados se requiere sustituir las ecuaciones (13) y (14) en (3) y (4), se tiene [54]:

$$m(\dot{v}_y + v_x \dot{\psi}) = -2C_\alpha^r \left( \frac{V_y - \omega l_r}{V_x} \right) + 2C_\alpha^f \delta - 2C_\alpha^f \left( \frac{V_y - \omega l_r}{V_x} \right)$$

$$I_z \ddot{\psi} = l_f \left( 2C_\alpha^f \delta - 2C_\alpha^f \left( \frac{V_y - \omega l_r}{V_x} \right) \right) - l_r \left( -2C_\alpha^r \left( \frac{V_y - \omega l_r}{V_x} \right) \right)$$

Reacomodando términos:

$$\dot{v}_y = -2 \left( \frac{C_\alpha^r + C_\alpha^f}{m V_x} \right) V_y + 2 \left( \frac{C_\alpha^r l_r + C_\alpha^f l_f}{m V_x} - V_x \right) \dot{\psi} + \frac{2C_\alpha^f}{m} \delta \quad (15)$$

$$\ddot{\psi} = 2 \left( \frac{C_\alpha^r l_r - C_\alpha^f l_f}{I_z V_x} \right) V_y - 2 \left( \frac{C_\alpha^r l_r^2 - C_\alpha^f l_f^2}{I_z V_x} \right) \dot{\psi} + \frac{2C_\alpha^f l_f}{I_z} \delta \quad (16)$$

Recordando que de manera similar a la ecuación (2):

$$\dot{v}_x = a_x + v_y \dot{\psi} \quad (17)$$

Para complementar el modelo de espacio de estados con las ecuaciones planteadas anteriormente se debe entender que al ser un modelo de 3GDL se requiere determinar la aceleración deseada, para lo cual se tiene la siguiente planta que representa el comportamiento del controlador [17]:

$$\dot{v}_x = \frac{1}{\tau s + 1} \dot{v}_{xdes} \quad (18)$$

O bien:

$$\dot{v}_{xdes} = \dot{v}_x + \tau \ddot{v}_x \quad (19)$$

Donde el controlador usa la aceleración deseada como entrada de control y la aceleración real del vehículo es acompañada de un retraso con valor de  $\tau$ .

De (15), (16) y (17) y (19) se obtiene el siguiente espacio de estados [54][55]:



$$\begin{aligned}
 \frac{d}{dt} \begin{bmatrix} v_y \\ \psi \\ \dot{\psi} \\ v_x \\ \dot{v}_x \end{bmatrix} &= \begin{bmatrix} -\frac{2C_\alpha^f + 2C_\alpha^r}{mv_x} & 0 & -\frac{2C_\alpha^f l_f - 2C_\alpha^r l_r}{m V_x} - V_x & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ -\frac{2C_\alpha^f l_f - 2C_\alpha^r l_r}{I_z v_x} & 0 & -\frac{2C_\alpha^f l_f^2 - 2C_\alpha^r l_r^2}{I_z V_x} & 0 & 0 \\ 0 & 0 & 0 & 0 & -\frac{1}{\tau} \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} v_y \\ \psi \\ \dot{\psi} \\ v_x \\ \dot{v}_x \end{bmatrix} + \begin{bmatrix} \frac{2C_\alpha^f}{m} \\ 0 \\ \frac{2C_\alpha^f l_f}{I_z} \\ 0 \\ 0 \end{bmatrix} \delta \\
 &+ \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \frac{1}{\tau} \end{bmatrix} u + \begin{bmatrix} 0 \\ 0 \\ 0 \\ v_y \dot{\psi} \\ 0 \end{bmatrix} \quad (20)
 \end{aligned}$$

El espacio de estados anterior se utilizará como representación de la dinámica vehicular en Matlab, siendo que la entrada  $u$  es el comando de aceleración longitudinal del vehículo. Dado que las salidas de velocidad longitudinal y lateral están representadas en términos del eje coordenado del vehículo, resta obtener dichas salidas en términos de la posición global (plano XY, véase Figura 5.2.1), lo cual se logra por medio de las siguientes relaciones:

$$\dot{X} = V_x \cos \psi - V_y \sin \psi$$

$$\dot{Y} = V_x \sin \psi + V_y \cos \psi$$

Este modelo nos resultara muy útil siempre y cuando la fuerza en los neumáticos no este saturada y la variación de la velocidad longitudinal sea pequeña, por ejemplo: cuando el radio de curvatura del terreno a seguir sea lo suficientemente grande.

Cabe señalar que los parámetros utilizados en la implementación del modelo anterior fueron extraídos directamente de los valores por default definidos en el banco de pruebas de Matlab para un vehículo sedan, dichos valores se resumen en la siguiente tabla:

Parámetro	Símbolo	Valor	Unidades
Masa	$m$	1575	$kg$
Momento de inercia	$I_z$	2875	$mNs^2$
Coefficiente de rigidez lateral delantero	$C_\alpha^f$	19000	$N/rad$
Coefficiente de rigidez trasero	$C_\alpha^r$	33000	$N/rad$
Distancia del CG al eje delantero	$l_f$	1.513	$m$
Distancia del CG al eje trasero	$l_r$	1.305	$m$

Tabla 5.2. Parámetros del vehículo EGO utilizados en la implementación del modelo

### 5.3 Control de Crucero Adaptativo

Tal y como se revisó, el control de crucero adaptativo (*Adaptive Cruise Control, ACC*) tiene el objetivo principal de mantener una distancia segura entre dos vehículos, se puede considerar como una extensión del controlador de crucero. Por tanto, podemos listar dos funciones principales:

- Control de velocidad
- Control de espaciamiento

Así mismo es necesario que el ACC determine qué tipo de control es el adecuado dependiendo de la situación, esto es posible debido a las mediciones obtenidas de sensores como radar y LiDAR, con las que se obtiene la velocidad y distancia relativas entre ambos vehículos [17][56][57]. El esquema presentado en la Figura 5.3 nos muestra el flujo de trabajo en el que funciona el ACC, donde los bloques azules describen los eventos necesarios para llevar a cabo los objetivos de

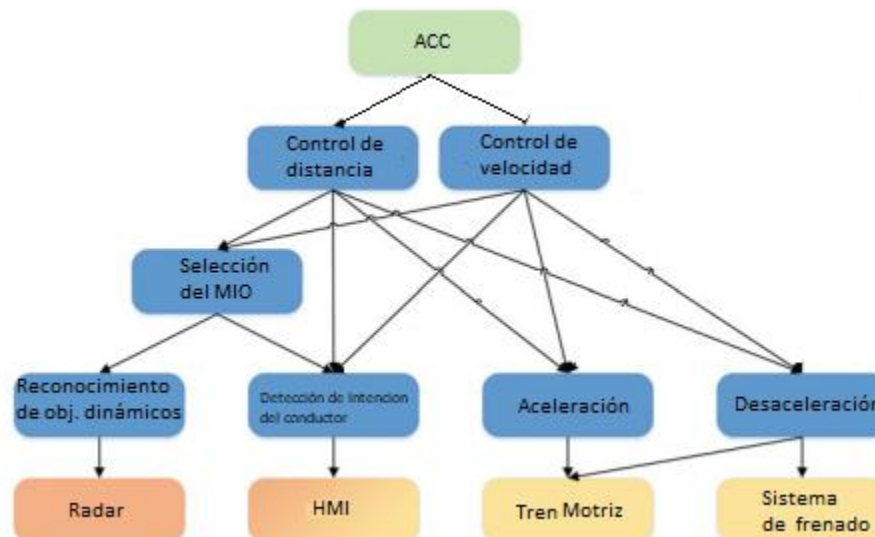


Figura 5.3. Diagrama del funcionamiento del ACC. Adaptado de [56]

control del ACC, mientras que los bloques amarillos describen a los actuadores y los naranjas a los sensores o entradas de información al sistema según corresponda.

#### 5.3.1 Arquitectura del controlador

Típicamente un ACC tiene una estructura que consta de dos partes con un acomodo jerárquico, es decir, un controlador superior y un controlador inferior. De tal manera que el controlador superior es aquel que determina la aceleración deseada para el vehículo. Posteriormente el comando de aceleración es enviado al controlador inferior, en donde se

implementarán los comandos necesarios para que los actuadores operen, ya sea acelerando o frenando dependiendo el caso [17][56].

En la Figura 5.3.1 se aprecia la arquitectura del sistema, en donde las entradas al nivel superior son proporcionadas por el conductor y el radar, tales como la distancia relativa, velocidad relativa y la velocidad a la que el conductor desea que el vehículo este en movimiento. En cuanto al nivel inferior se determinará mediante un switch lógico dependiendo del valor de la aceleración, dando como salida si el vehículo frenará o acelerará. Posteriormente, con el modelo propuesto con anterioridad se determinará tanto la posición como la velocidad del vehículo.

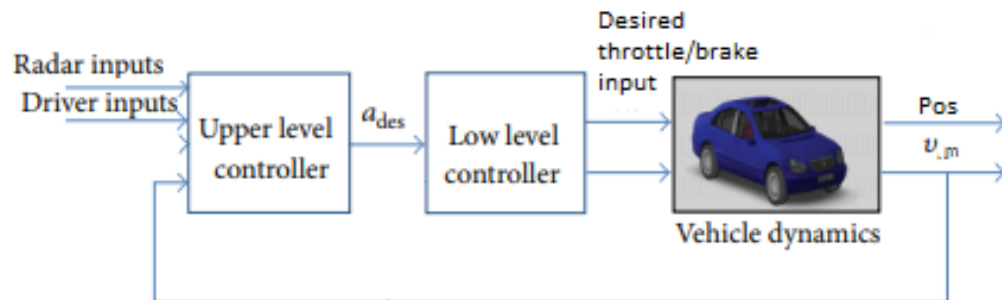


Figura 5.3.1. Arquitectura del Control de cruceo adaptativo [56]

## 5.4 Estrategias de control

El sistema analizado en este estudio es un modelo de dos vehículos, en el cual el vehículo con el sistema ACC se le denominará como EGO, mientras que el vehículo que lo precede será el vehículo MIO (*most important object*). También, se hace hincapié en que el estudio se centra en el entendimiento y evaluación del nivel superior del controlador, por lo que se utilizarán las siguientes cuatro estrategias de control [58]:

- Control Proporcional, Integral, Derivativo (PID)
- Control por modos Deslizantes (SMC, *sliding mode controller*)
- Control por constante de tiempo y espacio (CTG, *constant time-gap controller*)
- Control de modelo predictivo (MPC, *model predictive controller*)

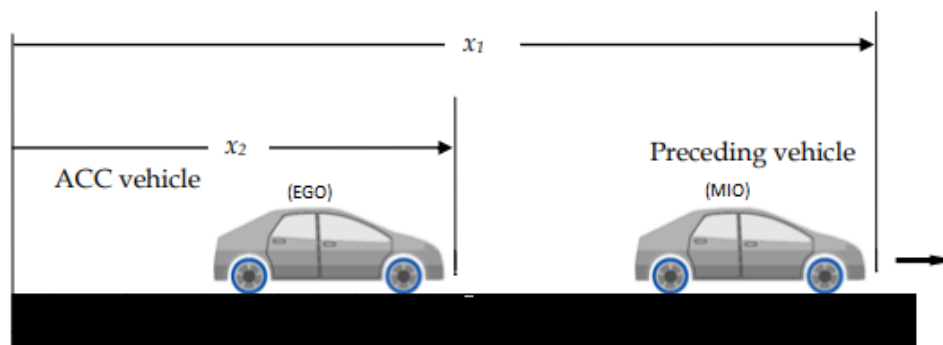


Figura 5.4. Sistemas de dos vehículos para analizar el ACC [58]

Para cada una de ellas se presentó el modelo teórico y su implementación mediante bloques en Simulink, posteriormente se realizó el uso de la metodología descrita en el capítulo 7 y los resultados individuales de cada uno en el banco de pruebas virtual; así como, la selección de parámetros reportada a lo largo del Capítulo 8 de este estudio.

### 5.4.1 Control Clásico (PID)

El controlador PID es de los más utilizados hoy en día debido a que su implementación es relativamente fácil. El controlador PID tiene la estructura mostrada en la Figura 5.4.1.1, para comprender este controlador se debe recordar que el objetivo de control es que el error en la velocidad del vehículo en estado estacionario sea cero.

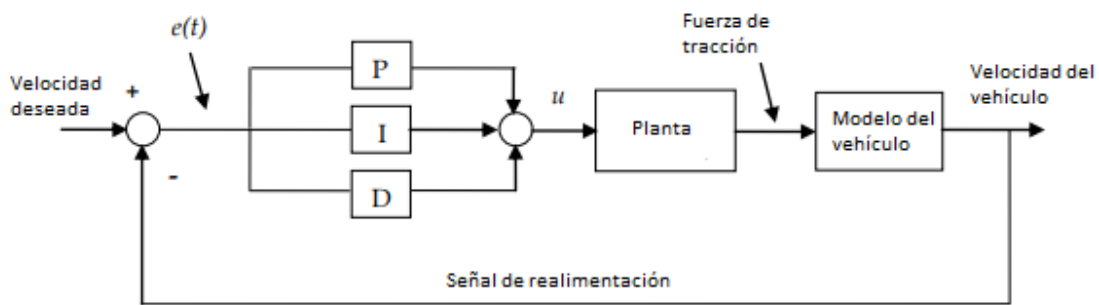


Figura 5.4.1.1. Diagrama de bloques para el sistema de control PID con señal de realimentación. [58]

Continuando con el diseño de este controlador, nuestra señal de control resulta ser función del error, específicamente es la suma del error proporcional, integral y derivativo, multiplicados por su respectiva constante (ecuación 21).

$$u = K_p e(t) + K_i \int e(t) dt + K_d \frac{de(t)}{dt} \quad (21)$$

La señal de error en este caso es la diferencia de velocidad entre la velocidad del vehículo y la velocidad deseada. Cada una de las constantes del controlador aporta una función al controlador. La constante proporcional se encarga de mejorar el tiempo de levantamiento; sin embargo, no elimina el error en estado estacionario. Por otro lado, la constante integral elimina dicho error sacrificando la respuesta en estado transitorio. Para corregir lo anterior la constante derivativa mejora la estabilidad general del sistema. Dichas constantes están estrechamente

relacionadas, el método utilizado para encontrar su valor fue el método de Ziegler-Nichols. Por lo que la ley de control PID aplicada específicamente al ACC es:

$$u = K_p(v_1 - v_2) + K_i(x_1 - x_2 - (L_{def} + hv_2)) + K_d(a_1 - a_2) \quad (22)$$

Donde  $K_p$ ,  $K_i$  y  $K_d$  son las ganancias del controlador, mientras que  $L_{def}$  es la distancia deseada entre el vehículo EGO y MIO por default. Así mismo,  $x_1, x_2, v_1, v_2, a_1$  y  $a_2$  son las posiciones, velocidad y aceleración del vehículo EGO y MIO respectivamente. El término  $hv_2$  es conocido como “*headway spacing policy*”, la cual corresponde a una variación del espacio entre el vehículo EGO y MIO, que depende proporcionalmente del tiempo  $h$ . La suma de  $L_{def}$  y  $hv_2$  dan como resultado la distancia segura  $L_{seg}$  que el vehículo EGO seguirá al momento de calcular el error. [17][58]

$$L_{seg} = L_{def} + hv_2 \quad (23)$$

Como todo modelo matemático se requiere poner a prueba para obtener los resultados deseados. Dicho lo anterior, se ha modelado el controlador mediante el uso de Simulink obteniendo la representación en diagrama de bloques inferior en la Figura 5.4.1.2.

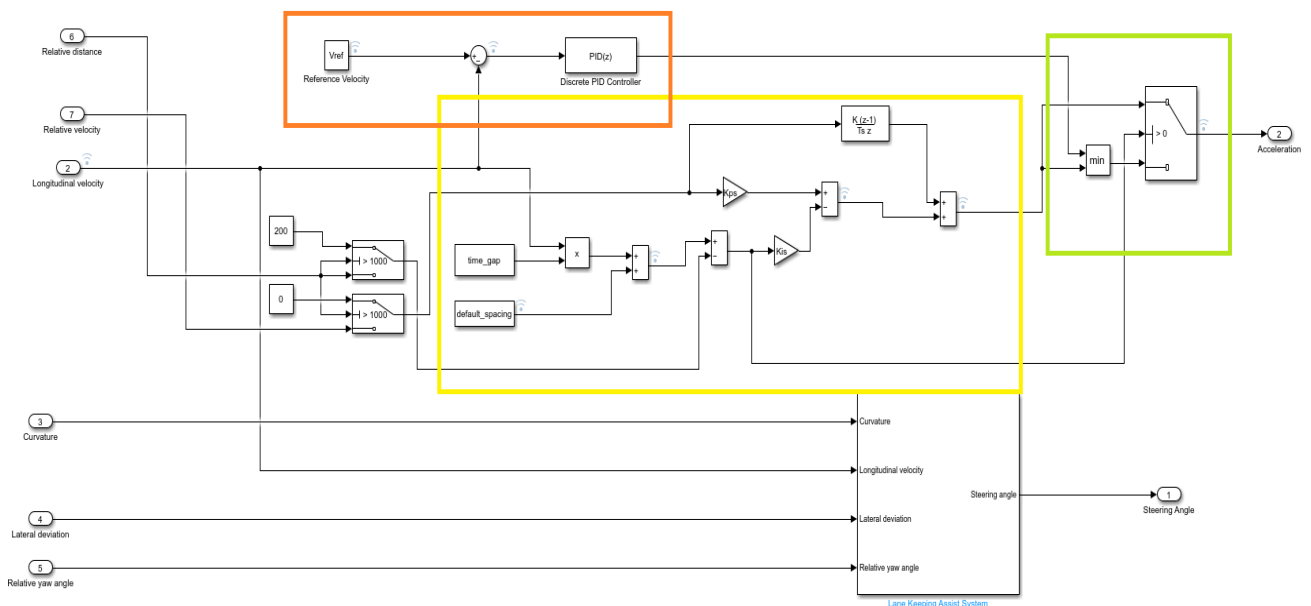


Figura 5.4.1.2. Implementación de controlador PID para el control superior de un ACC

En donde la zona enmarcada en color naranja corresponde al control de cruce de velocidad, la zona amarilla representa el modelo propuesto en la ecuación 22 y la zona en verde es el switch propuesto para determinar el momento para utilizar el control de velocidad o el de espaciado. Más detalles sobre su uso y selección de parámetros se presentan en el capítulo 8.

### 5.4.2 Constant Time-Gap Law

Otra manera apropiada para elaborar el análisis y control de un sistema ACC es a través del diseño de la ley de control mediante una constante de tiempo y espacio (CTG, por sus siglas en inglés). En este caso el control CTG también define la distancia entre los vehículos como una variación lineal con la velocidad, dado que se multiplica proporcionalmente por un parámetro constante, nuevamente el llamado “*headway spacing policy*” (ecuación 23). La diferencia con respecto al mencionado controlador PID es que señal de control depende directamente del error de la velocidad y distancia entre el vehículo EGO y MIO, tal y cómo se describe a continuación:

$$u = -\frac{1}{h}(\dot{\varepsilon} + \lambda_{CTG} \delta) \quad (24)$$

Donde:

$$\dot{\varepsilon}(t) = v_2 - v_1 \quad (25)$$

$$\delta(t) = x_2 - x_1 + L_{def} + hv_2 \quad (26)$$

De la ecuación 24 se define a  $u$  como la entrada y señal a controlar, es decir, el comando de aceleración deseada. El parámetro  $\varepsilon$  se refiere al rango existente entre ambos vehículos, mientras que  $\delta$  es el error en la distancia que existe entre los vehículos, este error está precedido por una constante  $\lambda_{CTG}$ , en este caso es un factor de peso entre los términos de la ecuación 24. Los parámetros de diseño en este caso particular son las constantes  $h$  y  $\lambda_{CTG}$ , las cuáles proporcionan estabilidad individual y colectiva al seguir un vehículo [17][58].

Si se sustituyen las ecuaciones 25, 26 en 24, se obtiene el siguiente modelo de diagrama de bloques en Simulink.

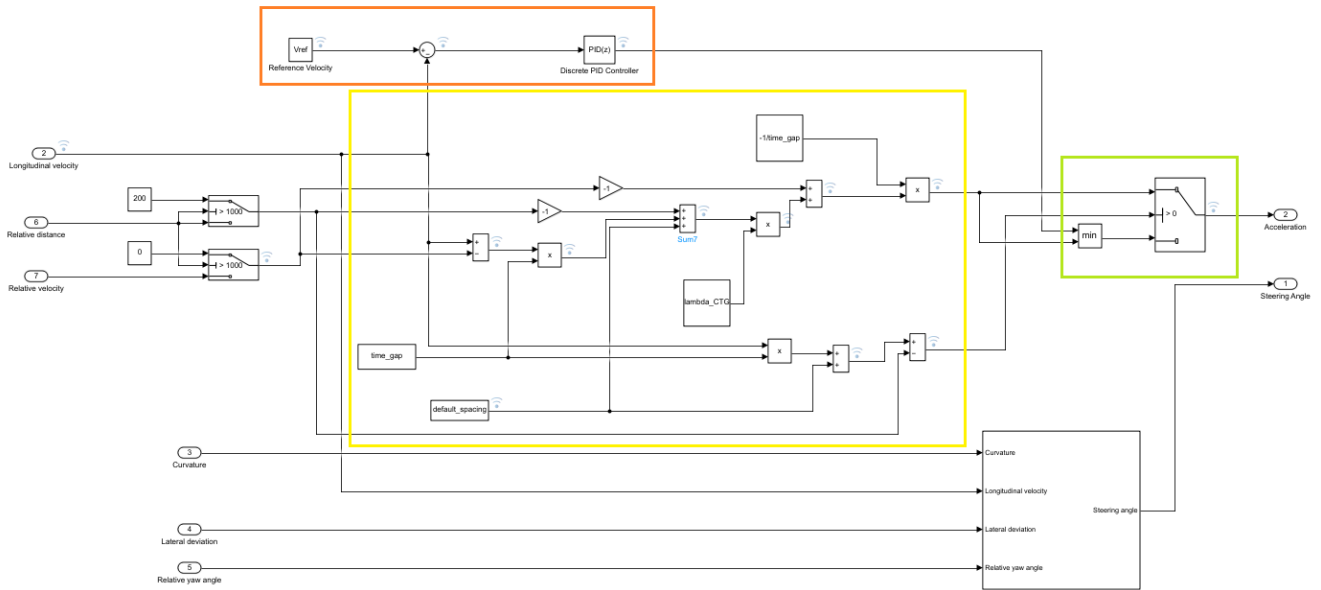


Figura 5.4.2. Implementación de la ley de control CTG para el control superior de un ACC

Siguiendo la misma secuencia que en el control PID, el control CTG está enmarcada por color amarillo, en el Capítulo 8 de este escrito se describe la selección de parámetros y análisis ante los escenarios propuestos.

### 5.4.3 Sliding Mode Control

Este tipo de control es utilizado principalmente en sistemas donde la planta objetivo es compleja y opera ante ciertas condiciones, es utilizado para sistemas no lineales y sistemas que son discontinuos.

Este método de control transforma un sistema de orden superior en un sistema de primer orden, con el propósito de ejecutar un control simple. Por lo tanto, en este sentido el control SMC depende del cambio de estado del sistema a una alta frecuencia, muy similar a como funcionaría un switch [58][59].

En el caso particular de nuestro sistema para un ACC se ha utilizado, de manera similar al PID y CTG, la ecuación 23 para definir nuestro objetivo de control. Por tanto, el error de la distancia entre los vehículos se expresa de la siguiente manera:

$$e = (L_{def} + hv_2) - (x_1 - x_2) \quad (27)$$

Donde el segundo término representa la distancia relativa entre los vehículos. Mediante el control SMC se pretende llevar este error a cero, y por tanto es seleccionado como una variable de deslizamiento  $S$  (*sliding variable*).

$$S = e = (L_{def} + hv_2) - (x_1 - x_2) \quad (28)$$

La ley de control debe de satisfacer la llamada condición de llegada definida a continuación:

$$\dot{S} \leq -\eta|S| \quad (29)$$

Donde  $\eta$  es una constante positiva

La ecuación 29 muestra que una señal de control adecuada debe garantizar que se cumpla la siguiente condición con la variable de deslizamiento:

$$\dot{S} = -\eta \operatorname{sgn}(S) \quad (30)$$

En la cual  $\eta$  debe ser lo suficientemente grande para cumplir la condición de la ecuación 29. Al cumplir este requisito significa que se ha modificado la trayectoria de nuestra variable de control en la superficie de deslizamiento definida como  $S = 0$ , es decir, se requiere forzar a que la variable de deslizamiento llegue a cero en un tiempo finito.

Diferenciando la ecuación 28 con respecto al tiempo se obtiene:

$$\dot{S} = hu - (v_1 - v_2) \quad (31)$$

Sustituyendo la ecuación 30 en 31 se obtiene la aceleración que debe de tener el vehículo EGO para cumplir la ecuación 29, esto implica que nuestra señal de control será [58][59]:

$$u = \frac{1}{h} (-\eta \operatorname{sgn}(S) - v_2 + v_1) \quad (32)$$



Donde  $u$  es la señal de control que se empleó para determinar la respuesta de nuestro sistema ACC, la Figura 5.4.3 muestra el resultado de implementar dicho modelo en Simulink.

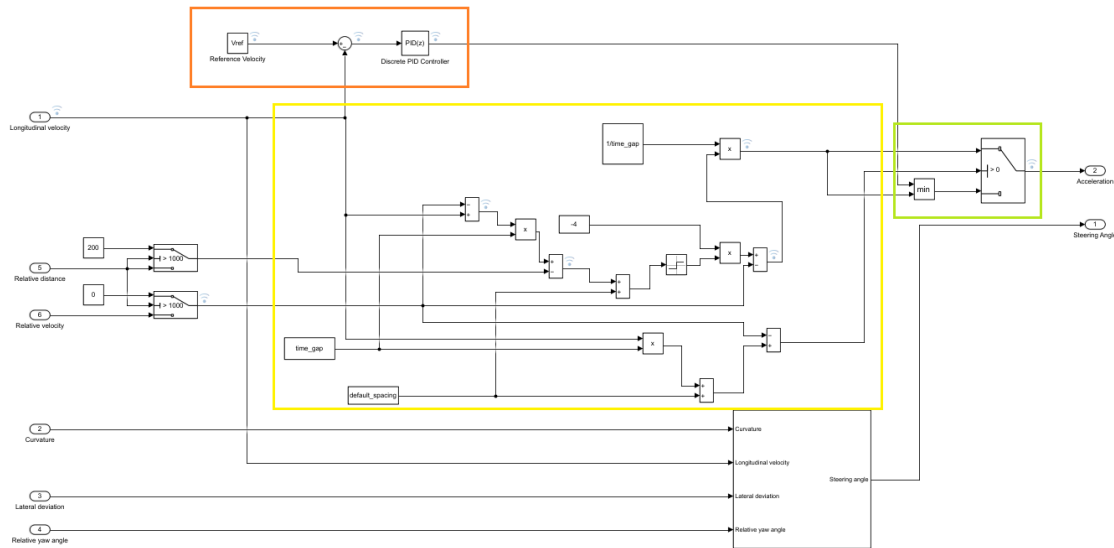


Figura 5.4.3. Implementación del controlador SMC para el control superior de un ACC

El recuadro amarillo es la representación de la ecuación 32 del SMC, mientras que el recuadro naranja y verde, representan el control de velocidad simple y el switch lógico que intercala ambos controladores. En la sección 8 se presenta la selección de nuestros parámetros de diseño,  $\eta$  y  $h$ .

### 5.4.4 MPC

Control predictivo por modelo (conocido por sus siglas en inglés como MPC, *Model Predictive Control*), es una técnica de control avanzada en donde el comportamiento del sistema de interés es predicho con el propósito de generar una entrada optimizada en la señal de control.

Algunas diferencias con respecto a los controladores anteriores son que es capaz de desarrollar un modelo de predicción para determinar las salidas futuras en instantes de tiempos (horizonte) y es capaz de generar una trayectoria de referencia, la cual tratará de seguir el controlador; también, realiza un cálculo de la secuencia del controlador para minimizar la función objetivo. Esta estrategia de control, hace uso de modelo de predicción basado en eventos pasados y actuales con el propósito de predecir las salidas futuras de la planta del sistema.

La estructura básica de la estrategia de control de un MPC se presenta en la Figura 5.4.4.1. Para comprender cómo funciona un MPC se deben tomar en cuenta los siguientes conceptos: la ventana de horizonte y el control de horizonte.

La ventana de horizonte se refiere al intervalo de tiempo en el que las salidas futuras serán predichas y su tamaño permanecerá constante, para lograr esto se debe tomar un tiempo  $t_i$  hasta un tiempo  $t_i + N_p$  para  $i = 1, \dots, N_p$ , en donde  $N_p$  es el horizonte de predicción, es decir, el tiempo máximo en el cual se realizará el cálculo del estado futuro.

El control de horizonte se caracteriza por asumir que el sistema opera durante un tiempo discreto, el tiempo actual se encuentra en el momento  $t$ , a partir de este punto, se derivan dos referencias, en adelante se encuentran los estados futuros del sistema y por el contrario antes del momento  $t$  se encuentran los estados pasados. Además de esto, se establece una trayectoria objetivo para el sistema a seguir (*Set-Point Trajectory*). Además, se define un camino de referencia (*Reference Path*) que comienza a partir de la salida en el tiempo  $t$  y define el camino ideal que la planta debería seguir hasta llegar a la trayectoria objetivo.

El control MPC hace uso de un modelo interno con el cual es capaz de predecir el comportamiento de la planta a partir del momento  $t$ . Mediante el uso de la información de la salida en el momento  $t$  y la señal de control futura  $u(t + m)$  para  $m = 0, \dots, N_c$  se pueden obtener las salidas futuras del sistema  $y(t + m)$  para  $m = 0, \dots, N_c$  hasta el límite propuesto con el horizonte de predicción. Donde  $N_c$  es el límite de pasos en los que se determinará la señal de control futura.

Una vez que la señal de control es calculada en el futuro solo el primer elemento del conjunto de estas señales es utilizado como señal de entrada a la planta del sistema, posteriormente el proceso de obtener la salida, predecir la señal futura y determinar la señal a utilizar se repite en un ciclo, por lo que en todo momento cada uno de las características que se han mencionado hasta el momento se actualizan con la información actual del sistema. [58]

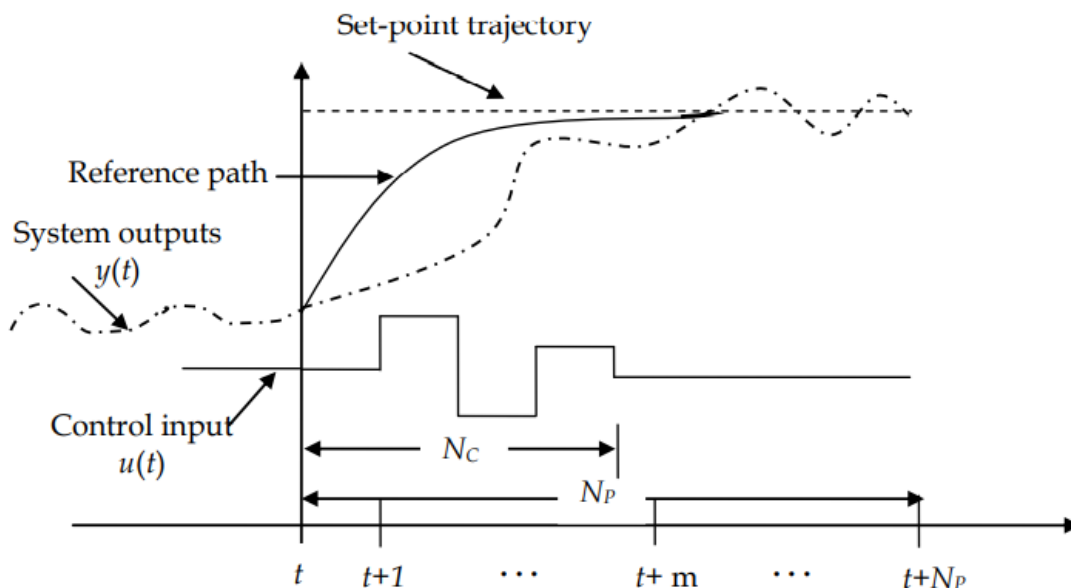


Figura 5.4.4.1. Estrategia de control de un MPC [58]

En el caso de nuestro sistema para el desarrollo del ACC se debe recordar la ecuación 19, la cual es la planta utilizada para el análisis del controlador superior, si se reescribe dicha ecuación en términos de la posición se obtiene lo siguiente:

$$u(t) = \tau \ddot{x}(t) + \dot{x}(t) \quad (33)$$

Donde  $u$  es la entrada de control,  $\tau$  el retraso entre el controlador superior y el controlador inferior,  $x$  la posición el vehículo EGO. Reescribiendo la ecuación 33 en un modelo de estados se obtiene:

$$\begin{pmatrix} x(t+T) \\ \dot{x}(t+T) \\ \ddot{x}(t+T) \end{pmatrix} = \begin{pmatrix} 1 & T & 0 \\ 0 & 1 & T \\ 0 & 0 & 1 - \frac{T}{\tau} \end{pmatrix} \begin{pmatrix} x(t) \\ \dot{x}(t) \\ \ddot{x}(t) \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ \frac{T}{\tau} \end{pmatrix} u(t) \quad (34)$$

Donde  $T$  es el tiempo de muestreo. La representación en modelo de estados nos proporciona información sobre los estados longitudinales del vehículo EGO.

Retomando nuestros objetivos de control se busca establecer una distancia segura entre ambos vehículos manteniendo una velocidad adecuada, para ello una vez más se hace uso de la ecuación 23.

Para poder determinar el movimiento relativo entre el vehículo EGO y MIO, es necesario representar el problema a resolver, para ello este se presenta en la Figura 5.4.4.2, en donde  $e$  es el error entre la distancia deseada y la distancia relativa  $R$ , mientras que  $L_{seg}$  es la distancia segura establecida por la ecuación 23.

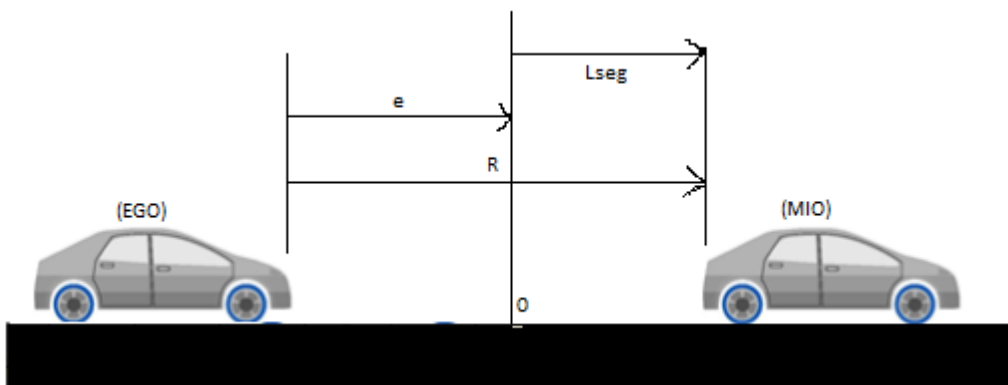


Figura 5.4.4.2. Distancia segura entre vehículos [58]

Utilizando la información de la Figura 5.4.4.2 se tiene que el error entre los vehículos se define como:

$$e_{K+1} = Ae_K + Bu_K \quad (35)$$

$$y_K = Ce_K \quad (36)$$

Donde:

$$e_K = \begin{pmatrix} err_k \\ \dot{err}_k \\ \ddot{err}_k \end{pmatrix} = \begin{pmatrix} -(R - L_{seg}) \\ \dot{R} \\ \ddot{x}_k \end{pmatrix} \quad (37)$$

Donde  $err_k$  es el error de espaciamento,  $\dot{err}_k$  es el error en la velocidad entre los vehículos y  $\ddot{err}_k$  es la aceleración absoluta del vehículo EGO. Para cada uno de estos elementos se busca que su valor sea cero. De la ecuación 35,  $u_K$  es la entrada de control, y  $y_K$  es la salida del sistema para el momento  $k$ . Al comparar las ecuaciones 35 y 36 con nuestro modelo de estados (ecuación 34) se tiene que las matrices A, B se definen como:

$$A = \begin{pmatrix} 1 & T & 0 \\ 0 & 1 & T \\ 0 & 0 & 1 - \frac{T}{\tau} \end{pmatrix}; \quad B = \begin{pmatrix} 0 \\ 0 \\ \frac{T}{\tau} \end{pmatrix} \quad (38)$$

De la literatura en [58] se tiene que la matriz C está determinada por:

$$C = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \end{pmatrix} \quad (39)$$

Como se describió anteriormente para un MPC y haciendo uso de la información medida se obtiene el vector del error en la información  $e_K$ , y posteriormente, se predicen los errores futuros hasta el horizonte de predicción ( $N_p$ ) determinado para cada momento del tiempo denominado  $k_i$ . Por tanto, el error futuro se define como:

$$e(k_i + 1 | k_i), e(k_i + 2 | k_i), \dots, e(k_i + m | k_i), \dots, e(k_i + N_p | k_i) \quad (40)$$

Donde el término  $e(k_i + m | k_i)$  corresponde al error que se predice utilizando la información del error actual  $e(k_i)$ . De manera similar se calcula el conjunto de señales de control futuras:

$$\Delta u(k_i), \Delta u(k_i + 1), \dots, \Delta u(k_i + N_c - 1) \quad (41)$$

En este caso  $\Delta u(k) = u(k) - u(k - 1)$ , esto implica que hablamos de un incremento en la señal con respecto al valor anterior, también se debe señalar que  $N_c$  será el factor que determine el número de parámetros usados para determinar la trayectoria futura. Se debe cumplir que  $N_c \leq N_p$ .

Una de las características más importantes de la estrategia de control del MPC es que genera una función de costo, conformada de los errores  $e_k$  y la señal de control que actúa, con lo que es capaz de determinar el mejor conjunto de entradas futuras para minimizar el error en la salida. Por tal razón, se puede encontrar la solución óptima para la entrada del controlador con la siguiente expresión:

$$\Delta U = (\Phi^T \Phi + \bar{R})^{-1} \Phi^T (R_s - F e_{ki}) \quad (42)$$

Donde las matrices  $\Phi, F$  y  $R$  son:

$$\Phi = \begin{bmatrix} CB & 0 & 0 & \dots & 0 \\ CAB & CB & 0 & \dots & 0 \\ CA^2B & CAB & CB & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{N_p-1}B & CA^{N_p-2}B & CA^{N_p-3}B & \dots & CA^{N_p-N_c}B \end{bmatrix} \quad (43)$$

$$F = \begin{bmatrix} CA \\ CA^2 \\ CA^3 \\ \vdots \\ CA^{N_p} \end{bmatrix} \quad (44) \quad ; \quad R_s = \begin{bmatrix} 1 \\ 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} r(k_i) , \quad r(k_i) = 0 \quad (45)$$

$R_s$  es el vector que contiene la trayectoria deseada a seguir y  $r(k_i)$  es el valor de dicha trayectoria en el momento  $k_i$ , dicho valor debe ser igual a cero ya que se busca que los términos del vector de error  $e_k$  sean cero. Por otro lado, la matriz  $\bar{R} = R I_{N_c \times N_c}$  ( $R \geq 0$ ) que corresponde a una matriz identidad cuya dimensión está determinada por el control de horizonte y es multiplicada por un término escalar de peso para la entrada del controlador, en el caso particular de este escrito este factor se debe considerar con un valor unitario  $R = 1$  [58].

En cada avance de tiempo  $k$ , el controlador MPC determinará la secuencia de entrada usando la ecuación 42. El modelo que se utilizó fue desarrollado en Simulink y se muestra en la Figura 5.4.4.3.

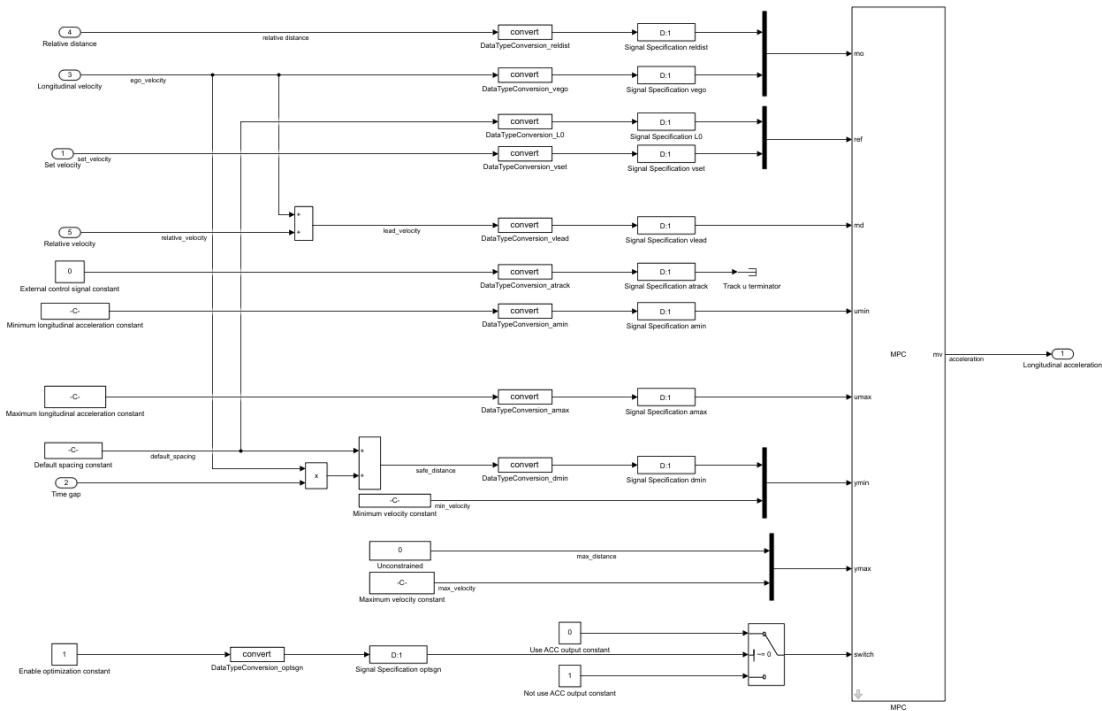


Figura 5.4.4.3. Implementación del controlador MPC para el control superior de un ACC [60]

Cabe aclarar que esta es la manera en la que opera el bloque de Matlab llamado “Adaptive Cruise Control System” incluido en su toolbox específico para conducción autónoma. [60] Dicho lo anterior, en este caso específico únicamente se determinaron los parámetros adecuados ante los escenarios descritos en el Capítulo 8, en donde se detallan estos y las restricciones que tiene el modelo para operar, es decir que el diagrama de bloques propuesto en la Figura 5.4.4.3. es el subensamble del bloque ACC desarrollado por Matlab.

Adicionalmente a los parámetros propios del controlador, se puede modificar el comportamiento del mismo mediante la variación de la robustez. En este contexto la robustez se refiere al hecho de modificar la velocidad de respuesta del controlador, por tanto, un menor valor producirá un controlador más robusto con acciones de control suaves, mientras que un valor mayor producirá un comportamiento de controlador más agresivo con un tiempo de respuesta más rápido. Esta evaluación se lleva a cabo en la Sección 8.1.1.

## 5.5 Conclusión

Se logró determinar el modelado dinámico del vehículo, así como su interacción y relación con cada una de las estrategias de control. Se revisaron cuatro diferentes estrategias de control para un mismo objetivo de control y desarrollar un sistema funcional del ACC. Se observó que las cuatro estrategias parten de la misma planta para el control superior del sistema; sin embargo, cada una de ellas determina de manera particular la forma en que el error se reducirá a cero. Así

mismo, la complejidad de cada una de ellas resulta ser un factor a considerar a la hora de implementarlas ya que cada una de ellas presenta características que pueden ser favorables o no a la hora de realizar la implementación del sistema ACC.





# 6. Banco de pruebas Virtual

Retomando la secuencia establecida en el capítulo 4, en esta sección se demostrará la composición del banco de pruebas/simulador que se utilizó a lo largo del estudio, se determinarán también los subsistemas que lo conforman y sus conexiones entre sí. Tal y como se determinó en el capítulo 3, la herramienta elegida para llevar a cabo la implementación del sistema ACC fue Matlab y Simulink, por tanto, todos los bloques que se presentan a continuación contienen información proveniente de dicho software.

## 6.1 Descripción del banco de pruebas

Matlab nos facilita el manejo y diseño de nuestro sistema ACC. Por tal razón, se partió de un programa preestablecido muy completo incluido en Matlab, cuyo nombre es “Highway Lane Following Test Bench” (HLFTB), el cual consiste en un banco de prueba que nos permite probar diferentes subsistemas, entre ellos la implementación del sistema ACC. En la Figura 6.1 se puede observar el diagrama de bloques del banco de pruebas en Simulink.

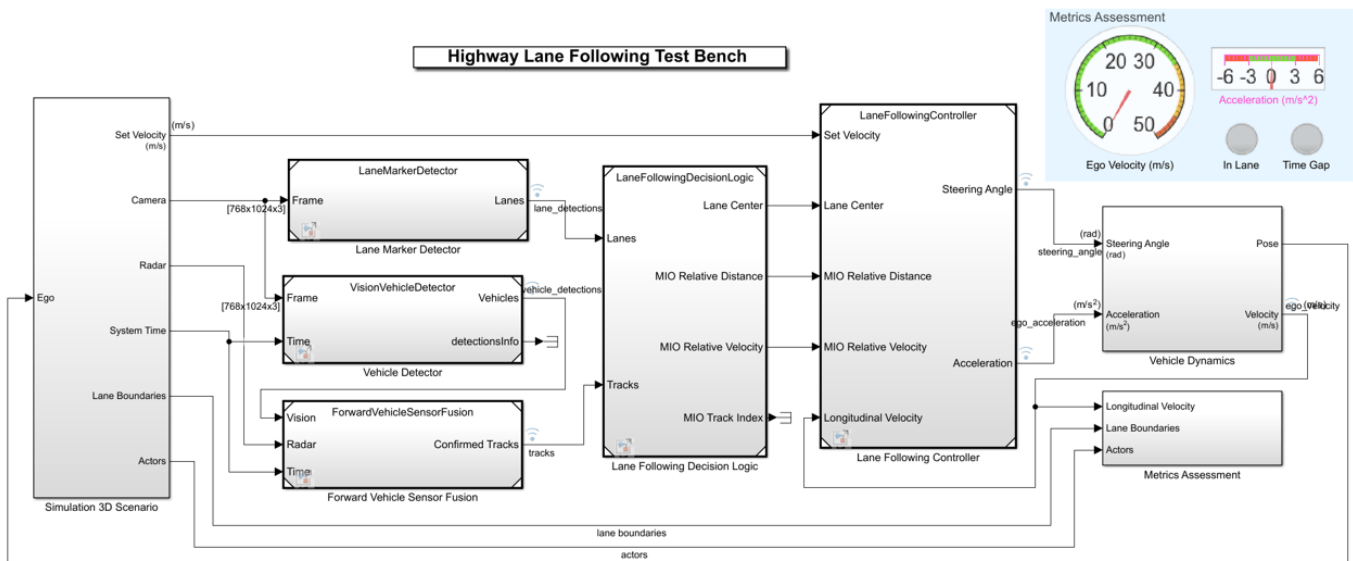


Figura 6.1. Modelo "Highway Lane Following Test Bench" extraído de Matlab [61]

Este banco de pruebas este compuesto por los siguientes elementos principales:

- *Simulation 3D Scenario*
- *Lane marker detector*
- *Vehicle detector*
- *Forward Vehicle Sensor Fusion*
- *Metrics Assesment*
- *Vehicle dynamics*
- *Lane following Logic and controller*

### **6.1.1 Simulation 3D Scenario**

Este subsistema es el encargado de especificar las condiciones de la pista, los actores o vehículos que aparecerán en los escenarios, así como los sensores utilizados.

Se relaciona con la descripción e implementación del escenario en sí y se compone de los bloques siguientes:

- **Scenario Reader:** Es el encargado de extraer las pistas y actores de un escenario previamente desarrollado. En síntesis, su función consiste en determinar las coordenadas del vehículo EGO y los actores en escena, así como determinar los límites del carril utilizado.
- **Vehicle to World:** Este bloque se encarga de convertir las coordenadas del vehiculo EGO en coordenadas globales de nuestro entorno.
- **Simulation 3D Scene Configuration:** establece el parámetro sobre el escenario a utilizar por default, que en este caso es una pista curva.
- **Cuboid To 3d Simulation:** este bloque transforma las coordenadas del vehículo ego en coordenadas del sistema en 3D.
- **Simulation 3D Camera:** Gracias a este bloque se captura la vista frontal del vehículo EGO para su posterior procesamiento en el subsistema Lane Marker Detector.
- **Simulation 3D Probabilistic Radar Configuration:** este bloque permite identificar vehiculos en el ambiente de simulación.

El sistema completo se puede visualizar en la Figura 6.1.1:

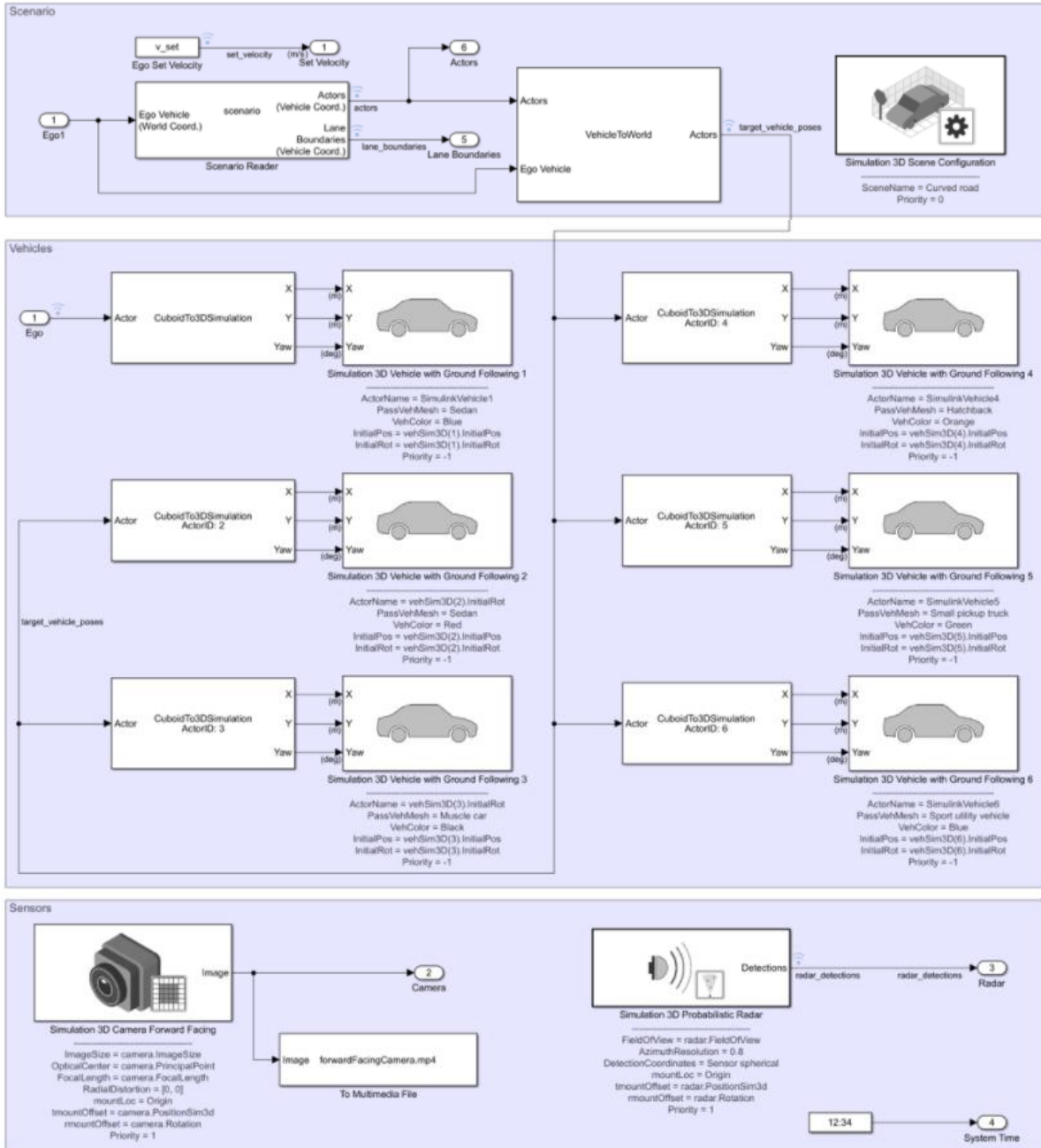


Figura 6.1.1. Subsistema Simulation 3D Scenario [61]

## 6.1.2 Lane Marker Detector

En este particular subsistema se hace uso de un algoritmo que implementa un módulo de percepción para analizar las imágenes de la pista. Funciona mediante la toma de una imagen desde la cámara, su propósito principal es determinar los límites del carril en el que el vehículo

está viajando. El bloque enviará la información del carril o carriles si detecta más de uno. Este subsistema se muestra en la Figura 6.1.2.

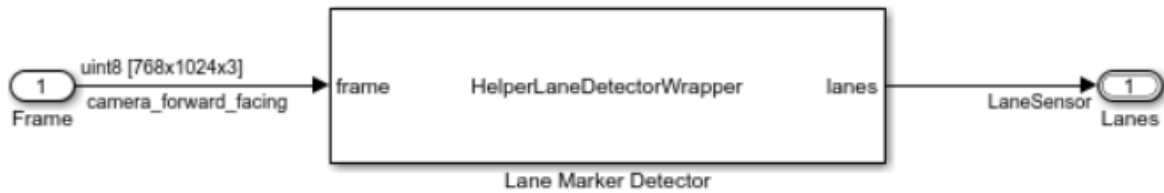


Figura 6.1.2. Subsistema Lane Marker Detector [61]

### 6.1.3 Vehicle Detector

Como su nombre lo indica este subsistema se hace cargo de identificar los vehículos en el escenario actual mediante el uso de la información capturada por la cámara, dicha información es procesada y enviada al siguiente subsistema en forma de cajas delimitadoras, véase Figura 6.1.3.

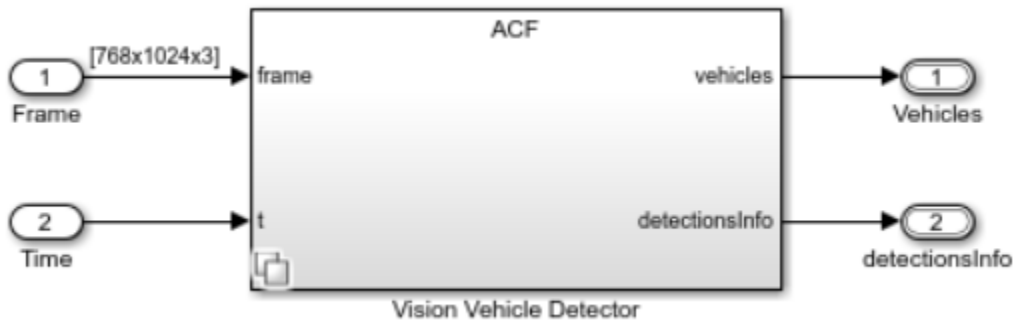


Figura 6.1.3. Subsistema Vehicle Detector. [61]

### 6.1.4 Forward Vehicle Sensor Fusion

Este Sistema une la información que proviene de la cámara y del sensor de radar con el propósito de detectar los vehículos que actúan en el escenario usando un método específico conocido como método de seguimiento por nivel central. Los elementos detectados por el radar son agrupados y concatenados con la información detectada por la cámara, una vez realizada esta comprobación los vehículos son seguidos usando el bloque Multi-Object Tracker, cuya función es la confirmación de los elementos y proveer esta información para el siguiente subsistema. Tal y como se visualiza en la Figura 6.1.4.

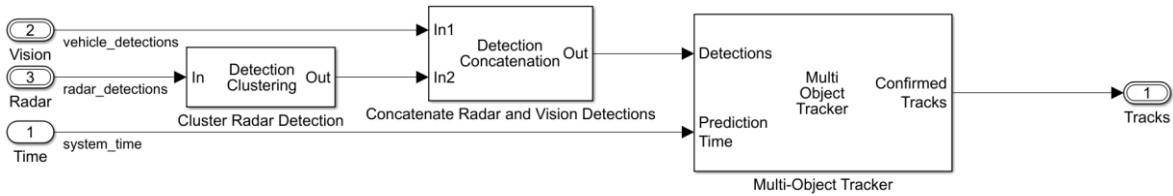


Figura 6.1.4. Subsistema Forward Vehicle Sensor Fusion. [61]

### 6.1.5 Lane Following Decision Logic

Este subsistema toma la información sobre los objetos detectados, así como información sobre los carriles presentes en el escenario. De esta manera es capaz de estimar el centro del carril y determinar cuál es el vehículo MIO (*most important object*) viajando en el mismo carril que el vehículo EGO. Como resultado, proporciona información relevante como la distancia relativa y velocidad relativa entre ambos vehículos, los cuales son parámetros que son utilizados más adelante por el controlador y el desarrollo de nuestro sistema ACC, Figura 6.1.5.

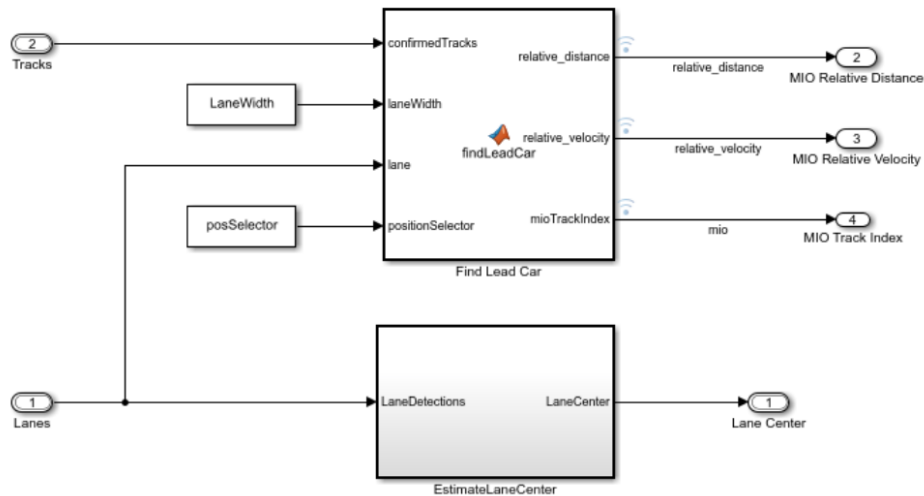


Figura 6.1.5. Sub sistema Lane Following Decision Logic [61]

### 6.1.6 Lane Following Controller

Es en este conjunto de bloques donde se toma la velocidad establecida para el control de velocidad y, por tanto, se requiere como entrada el estado actual de la velocidad longitudinal y la distancia entre ambos vehículos (EGO y MIO), además requiere de determinar el ángulo de yaw relativo a la curvatura del carril (*Preview Curvature*) como entrada al controlador para que este mismo pueda determinar el ángulo de giro de la dirección. Además, se realiza la implementación del sistema ACC descrito en la sección 5.4, ya que se reemplaza cada una de las estrategias de

control por el bloque inferior nombrado como *MPC Controller*; por tanto, el propósito principal de este bloque es determinar el comando de aceleración del sistema. Adicionalmente, se hace uso del bloque *Watchdog Braking Controller*, cuya función es aplicar un comando de frenado en caso de un fallo en el sistema para desacelerar el mismo. Ambas señales son evaluadas posteriormente para determinar el estado de la señal de aceleración, ya sea que el sistema indique que está acelerado, frenando o desacelerando. Dicho sistema se muestra en la Figura 6.1.6.

En resumen, la función principal de este subsistema es especificar el control lateral y longitudinal del vehículo EGO.

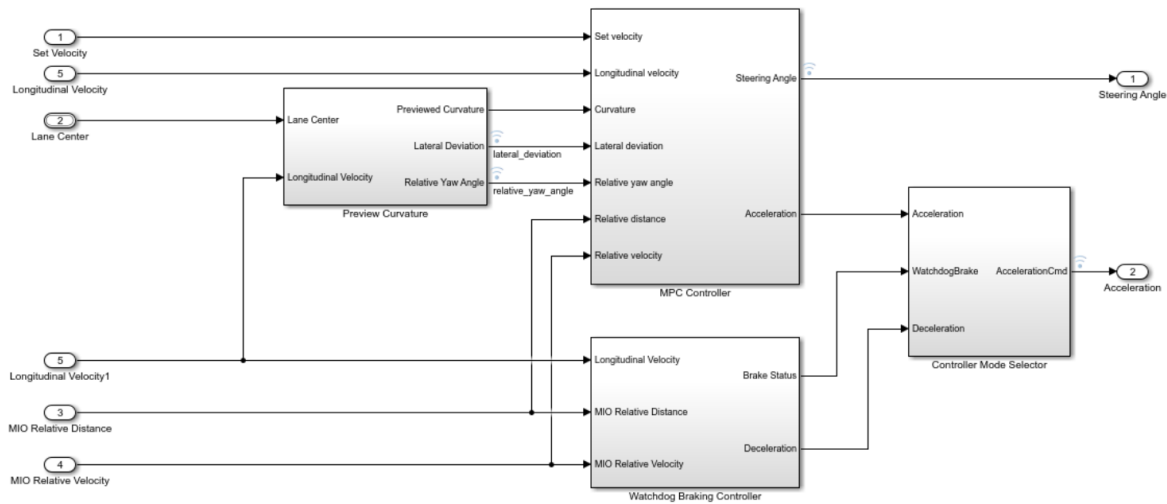


Figura 6.1.6. Subsistema Lane Following Controller. [61]

### 6.1.7 Vehicle Dynamics

Dentro de este subsistema se encuentra el modelo dinámico utilizado para representar nuestro vehículo, el bloque *Bicycle Model* proporciona dicha información, tal y como se presentó en la sección 5.2, se hace uso de un modelo de dos ruedas con 3 grados de libertad, el cual nos permitirá determinar información relevante con respecto al movimiento longitudinal, lateral y orientación (yaw). En cuanto al tren de potencia (controlador inferior del ACC), es presentado de manera trivial donde se toma en cuenta el comando de aceleración proporcionado por el controlador y este se descompone en la fuerza lateral y longitudinal necesarias para el modelo dinámico. Una vez procesada la información, las salidas del subsistema son la posición del vehículo EGO y su velocidad, Figura 6.1.8.

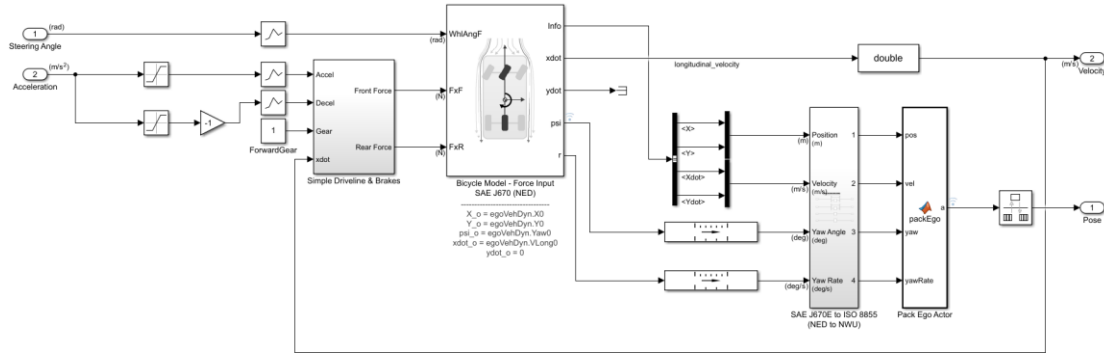


Figura 6.1.8. Subsistema Vehicle Dynamics. [61]

### 6.1.8 Metrics Assessment

El propósito de este subsistema es asegurar la evaluación de los métricos del sistema haciendo uso de la información real que se conoce y es provista por el escenario utilizado, para ello se evalúan cuatro métricos del sistema de seguimiento de carril:

- **Verify Lateral Deviation:** verifica la desviación lateral desde el centro de la línea del carril con ciertos límites dependiendo del escenario evaluado.
- **Verify Lane:** verifica que el vehículo EGO realmente este siguiendo alguno de los carriles de la pista durante toda la simulación.
- **Verify Time Gap:** se hace cargo de verificar que el tiempo de espaciado entre el vehículo EGO y el vehículo MIO sea mayor a 0.8 segundos.
- **Verify no Collision:** se verifica que el vehículo EGO no colisione con el vehículo seleccionado como MIO en ningún momento durante toda la simulación.

El diagrama de este subsistema se muestra en la Figura 6.1.8.

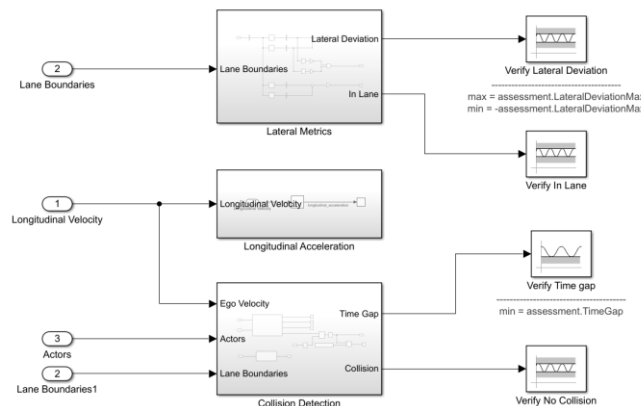


Figura 6.1.8. Subsistema Metrics Assessment. [61]

## **6.2 Conclusión**

A lo largo de esta sección se presentaron los subsistemas que conforman nuestro banco de pruebas virtual. Se puede observar inmediatamente la ventaja y utilidad de una plataforma de desarrollo como Matlab y Simulink, existen muchas herramientas que nos facilitan el desarrollo y unificación de este tipo de sistemas, de igual manera distintos bloques que hacen posible la implementación de este tipo de simulaciones. El sistema utilizado en esta sección es bastante amplio y por tanto puede ser modificado a las necesidades del usuario dependiendo el objeto de estudio, es por eso que en nuestro caso solamente centramos el desarrollo en la parte que conforma el controlador del ACC, el sistema en si es bastante personalizable, esto implica que podemos diseñar a cualquier nivel que sea requerido, desde un nivel conceptual, lógico o de implementación de cada subsistema.





# 7. Metodología

En este capítulo se abordará la metodología que se llevó a cabo para determinar los parámetros de cada una de las estrategias de control descritas en el Capítulo 5, se describirán los escenarios propuestos con base en los estándares internacionales ISO 15622 e ISO 22178; así como su implementación gracias a Matlab.

Lo que se pretende es llevar a cabo pruebas de una manera que sea eficiente y sistemática con la que podamos determinar los valores adecuados para que el sistema ACC funcione correctamente con cada uno de los controladores. Bajo esta premisa definiremos eficiente como la reducción de tiempo y costo de pruebas comparado con lo que tomaría si se realizarán pruebas físicas en un campo de pruebas para vehículos. Así mismo, el concepto de sistemático se definirá como el conjunto de pasos que se llevarán a cabo para diseñar cada controlador y el procedimiento de prueba para cada escenario propuesto.

## 7.1 Normatividad Internacional ISO 15622 e ISO 22178

Primeramente, se debe entender que son y que establecen estas normas de estándar internacional, se identifica como ISO 15622 *“Intelligent Transport Systems - Adaptive Cruise Control sistemas - Performance requirements and test procedures”* e ISO 22178 *“Low Speed Following (LSF) Systems Performance requirements and test procedures”*. Estas normas contienen la estrategia de control básica, los requerimientos mínimos para diagnóstico y reacción ante una posible falla, así como procedimientos para probar el rendimiento de sistemas de seguimiento de velocidad, tales como el control de crucero adaptativo.

En conjunto, y dentro de esta normatividad se enumeran varias pruebas importantes; sin embargo, dichas pruebas se pueden enumerar en los siguientes métodos de prueba principales que son utilizados para evaluar el desempeño de un sistema ACC:

- **Identificación del vehículo objetivo**
- **Prueba de desaceleración automática**
- **Prueba de reajuste de vehículo objetivo automático**
- **Prueba de capacidad en curva**

Considerando estos elementos de prueba, los escenarios utilizados y descritos en la siguiente sección cumplen dichas características y ponen a prueba el sistema por diseñar. Mayor información con respecto a las especificaciones de las normas estándares internacionales ISO 15622 e ISO 22178 se puede encontrar en las referencias. [62][63]

## 7.2 Escenarios/Casos de prueba

Para poder evaluar el desempeño de los controladores se realizaron pruebas en 6 escenarios diferentes en el banco de pruebas virtual, incluidas en la plataforma Matlab en conjunto con *Unreal Engine* como interfaz gráfica. Estos escenarios contienen una combinación de las características de la sección anterior y se describen a continuación:

- Escenario 1:** "scenario\_LFACC\_01\_Curve\_DecelTarget", este escenario contiene dos vehículos, el vehículo EGO (1) y el MIO (2). En primera instancia el vehículo EGO comienza con una velocidad inicial de 18 [m/s] hasta alcanzar una velocidad establecida deseada de 20 [m/s] cuando no hay presencia de ningún vehículo en frente del vehículo EGO. Paralelamente el vehículo MIO viaja a una velocidad constante de 22 [m/s] y después de 11 [s] reduce su velocidad hasta 18.5 [m/s] con una desaceleración de  $-1.7$  [m/s<sup>2</sup>]. Posteriormente, mientras ambos vehículos se encuentran en movimiento el EGO comienza a modificar su velocidad para respetar la distancia segura entre ambos vehículos. Finalmente, continúan a velocidad constante hasta que el vehículo MIO realiza un freno repentino y el EGO, siguiendo el mismo comportamiento, frena en un intervalo de tiempo determinado. Dicho comportamiento se observa en la Figura 7.2.1 y 7.2.2.

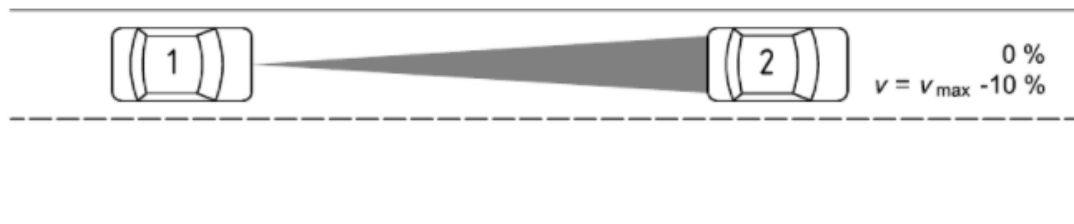


Figura 7.2.1. Escenario 1. Desaceleración Automática en una curva. [63]

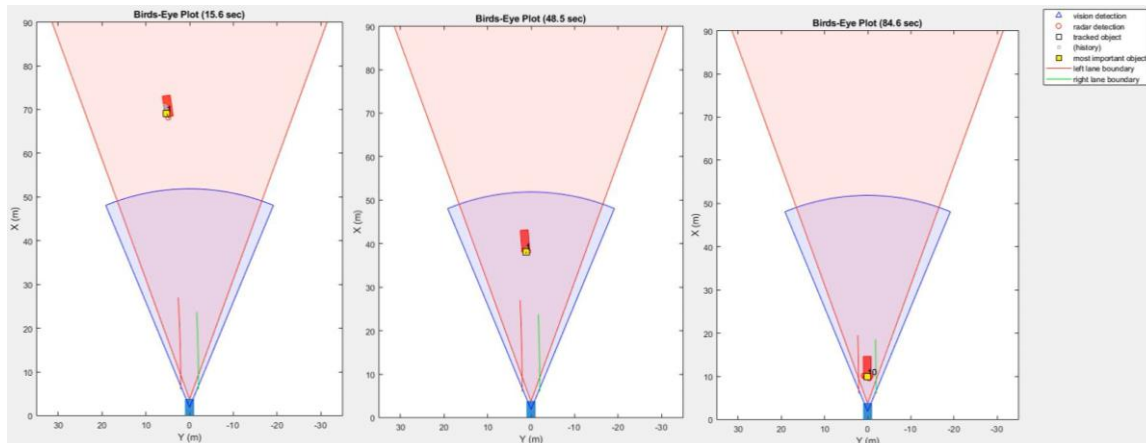


Figura 7.2.2. Representación del escenario 1 mediante un gráfico de tipo "Birds-Eyes" con tres momentos principales: izquierda: vehículo alejado a velocidad constante, centro: disminución de velocidad del vehículo EGO, derecha: alto total de ambos vehículos. [61]

Otra manera en la que se representa el escenario es mediante el uso del motor gráfico *Unreal Engine*, en este caso la misma escena que en la Figura dos se concatena con la Figura 7.2.3.

Esto es de gran utilidad ya que podemos observar el escenario de una manera más fiel a lo que podría suceder en la vida real.



Figura 7.2.3. Representación del escenario 1 mediante el uso de Unreal Engine. [61]

- Escenario 2:** “scenario\_LFACC\_02\_Curve\_AutoRetarget”, este escenario contiene tres vehículos, el vehículo EGO (1) y dos vehículos que lo anteceden, a los cuales denominaremos, vehículo 2 (inicialmente el MIO) y vehículo 3. Como punto de partida, el vehículo EGO viaja a una velocidad de 14 [m/s], mientras que el vehículo 2 a 13.9 [m/s] y el 3 a una velocidad constante de 10 [m/s]. Posteriormente el vehículo 2 realiza un cambio de carril para dejar al vehículo 3 atrás; sin embargo, es en este instante donde el vehículo EGO reajustará el vehículo objetivo y comenzará a disminuir su velocidad hasta igualar la velocidad del vehículo 3 (nuevo MIO). Esto se puede observar con detalle en la Figura 7.2.4.

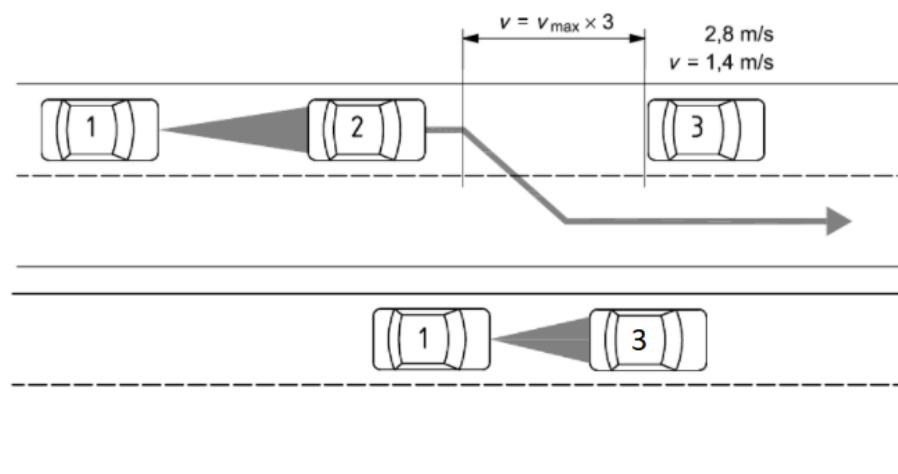


Figura 7.2.4. Escenario 2, Reajuste de objetivo automático después de cambio de carril en una curva. [63]

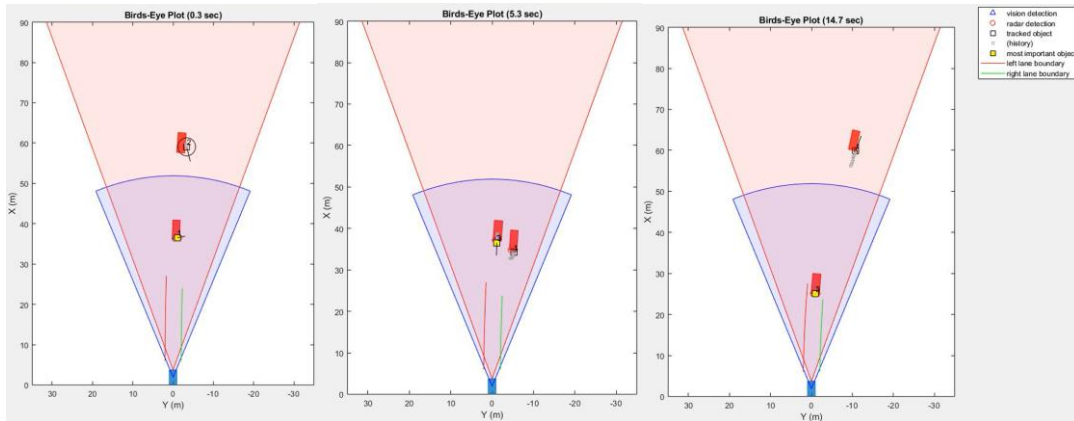


Figura 7.2.5. Representación del escenario 2 mediante un gráfico de tipo "Birds-Eyes" con tres momentos principales: izquierda: vehículo en línea recta, centro: cambio de carril del vehículo 2, derecha: reajuste de objetivo con el vehículo 2. [61]



Figura 7.2.6. Representación del escenario 2 mediante el uso de Unreal Engine. [61]

- Escenario 3:** "scenario\_LFACC\_03\_Curve\_StopnGo", este escenario contiene múltiples vehículos, la mayoría de estos circulan en los carriles adyacentes mientras el vehículo EGO y MIO interactúan en un camino curvo. Inicialmente el vehículo EGO viaja a una velocidad de 14 [m/s] al igual que el MIO. Después de transcurrir 4 [s] el vehículo MIO reduce su velocidad a 8 [m/s] con una desaceleración de 1.7 [m/s<sup>2</sup>] y mantiene dicha velocidad constante durante un tiempo de 10 [s]. Finalmente, el MIO aumenta su velocidad nuevamente a un valor de 13 [m/s], en donde se mantiene constante hasta finalizar la simulación. Durante los cambios de velocidad del MIO, el vehículo EGO debe ajustar su velocidad en un intervalo de tiempo corto para mantener una distancia segura entre ambos vehículos. De manera similar a lo que sucede en el escenario 1, hay una reducción de velocidad, con la principal diferencia que la reducción de velocidad sucede en un momento más corto y hay más vehículos en los carriles adyacentes, esto se puede observar en las Figuras 7.2.7 y 7.2.8.

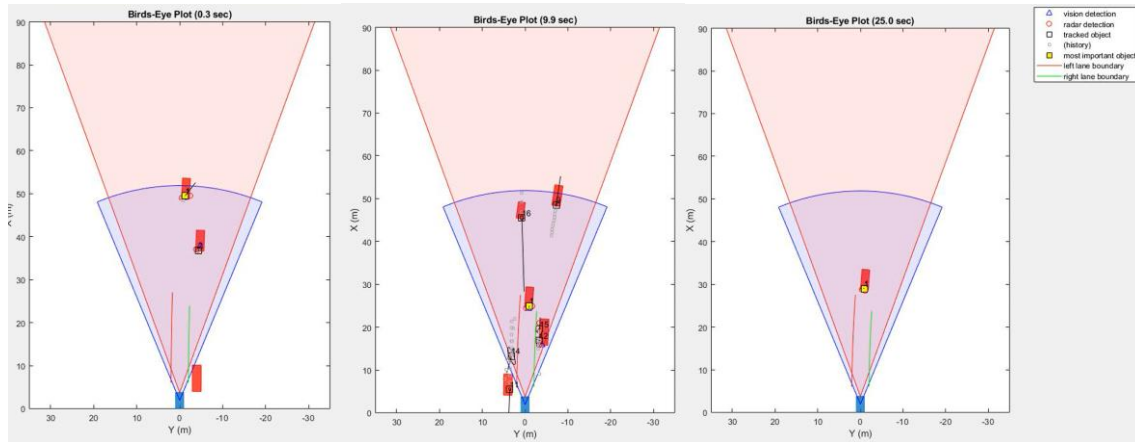


Figura 7.2.7. Representación del escenario 3 mediante un gráfico de tipo "Birds-Eyes" con tres momentos principales: izquierda: vehículo alejado a velocidad constante, centro: disminución de velocidad del vehículo MIO, derecha: aumento de velocidad vehículo MIO. [61]



Figura 7.2.8. Representación del escenario 2 mediante el uso de Unreal Engine.. [61]

- Escenario 4:** "scenario\_LFACC\_04\_Curve\_CutInOut", este escenario contiene 4 vehículos principales, el vehículo EGO (1), vehículo 3, 4 y 2. En principio, el vehículo EGO y el vehículo 3 se encuentran en el carril uno con una velocidad inicial de 20.6 [m/s] y 18 [m/s] respectivamente. Mientras tanto el vehículo 4 y 2 se encuentran en el carril 2 adyacente al carril uno con una velocidad de 11.1 [m/s] y 19.6 [m/s] respectivamente. Posteriormente el vehículo 2 realiza un cambio de carril del 2 al 1 adelantando así al vehículo 4 y frenando al vehículo EGO, después de unos segundos vuelve a cambiar de carril del 1 al 2 para adelantar por completo al vehículo 4. Durante este cambio el vehículo EGO ajusta su velocidad para evitar un choque con el vehículo 2 y posteriormente ajusta su velocidad nuevamente para mantener una distancia segura con el vehículo 3, ahora el vehículo MIO. Este evento se puede observar en la Figura 7.2.9.

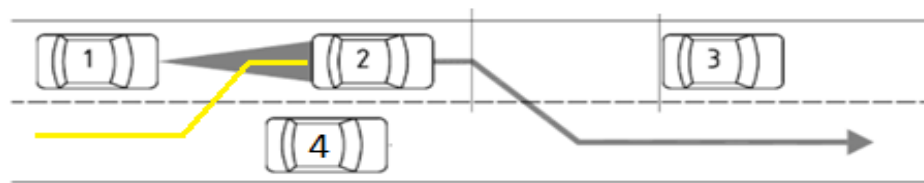


Figura 7.2.9. Escenario 4 y 5, Vehículo adyacente con doble cambio de carril en curva. Adaptado de [63].

- Escenario 5:** “scenario\_LFACC\_05\_Curve\_CutInOut\_TooClose”, este escenario se describe de la misma manera que el escenario 5, con la diferencia que el tiempo y distancia en que el vehículo 3 realiza el cambio de carril es menor. Por tal motivo el escenario 4 y 5 comparten la misma representación en Matlab, la cual se muestra a continuación en las Figuras 7.2.10 y 7.2.11.

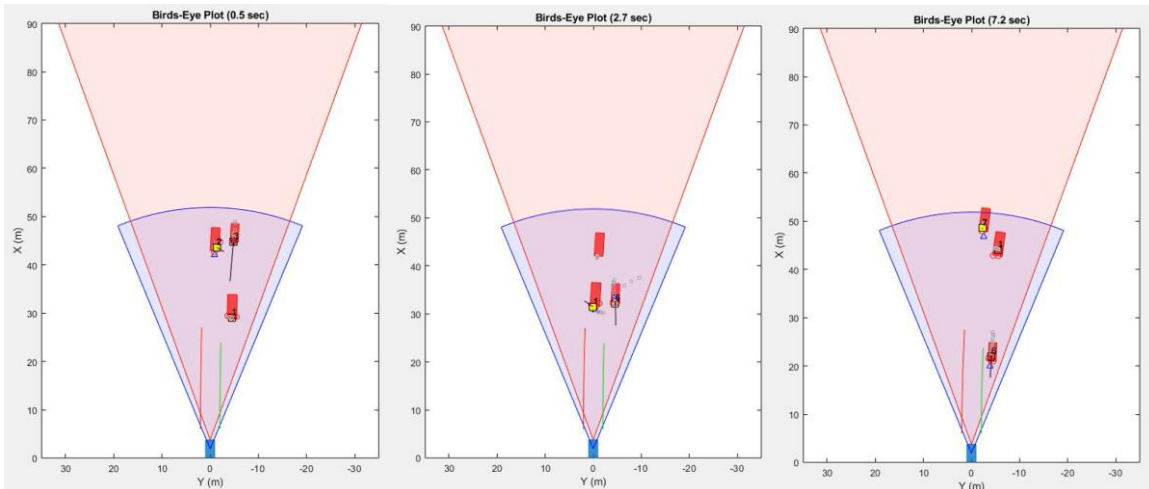


Figura 7.2.10. Representación del escenario 4 y 5 mediante un gráfico de tipo "Birds-Eyes" con tres momentos principales: izquierda: vehículos en estado inicial, centro: cambio de carril del vehículo 2, derecha: termino de cambio de carril y reajuste de objetivo. [61]



Figura 7.2.11. Representación del escenario 4 y 5 mediante el uso de Unreal Engine. [61]

- Escenario 6:** “scenario\_LFACC\_06\_Straight\_StopandGoLeadCar”, este escenario consta de dos vehículos, el vehículo EGO y el MIO, inicialmente el vehículo EGO comienza una velocidad de 25 [m/s] hasta encontrarse con el vehículo MIO que se encuentra totalmente detenido, posterior a ello este último comienza a acelerar hasta alcanzar una velocidad de 22 [m/s], durante este proceso el vehículo EGO mantendrá una distancia segura y alcanzará la velocidad deseada. De manera similar al escenario 1 el vehículo EGO realiza un frenado repentino; sin embargo, como se describe después de unos instantes continua su camino con respecto al vehículo MIO, Figura 7.2.12 y 7.2.13.

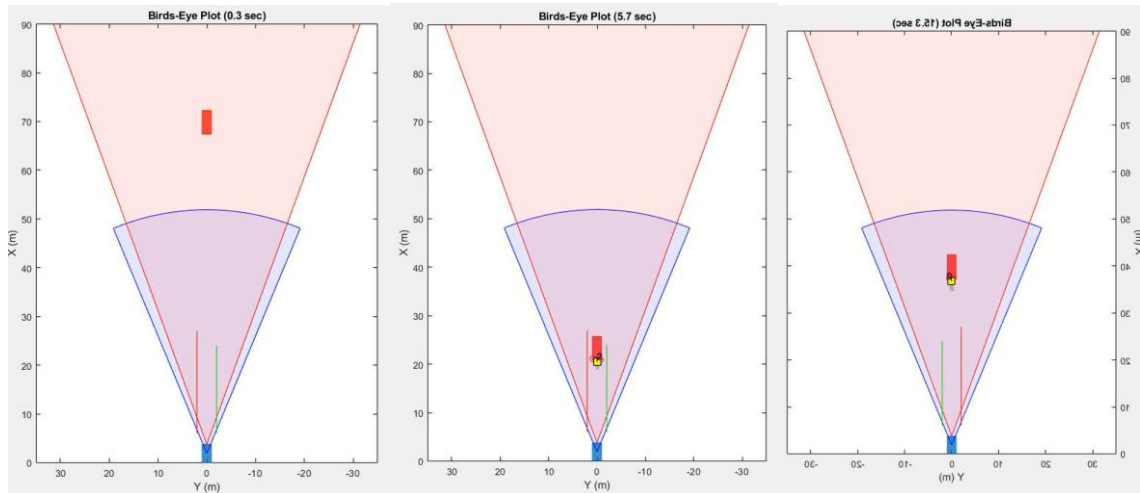


Figura 7.2.12. Representación del escenario 6 mediante un gráfico de tipo "Birds-Eyes" con tres momentos principales: izquierda: vehículo EGO a velocidad constante, centro: frenado de emergencia ante el vehículo MIO, derecha: avance a velocidad constante partiendo de alto total. [61]

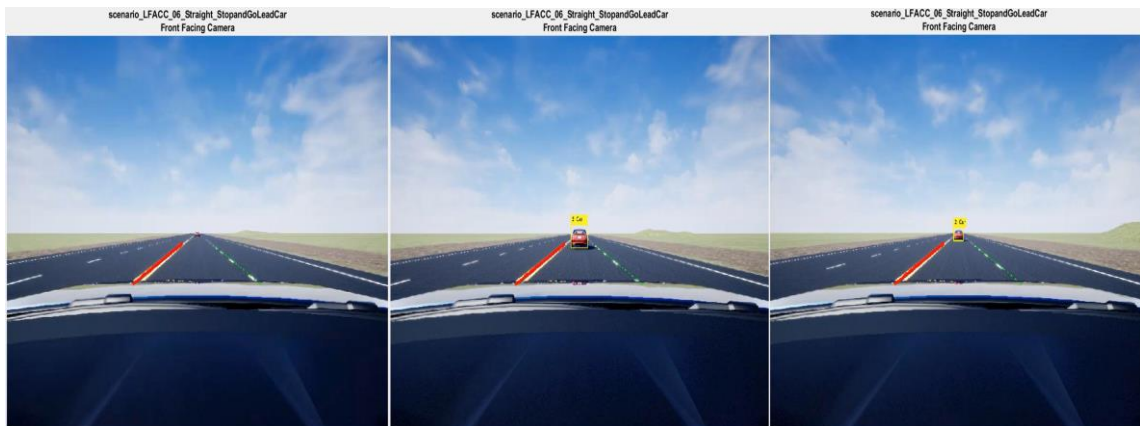


Figura 7.2.13. Representación del escenario 4 y 5 mediante el uso de Unreal Engine. [61]

### 7.3 Conclusión

En este apartado se ha planteado la metodología propuesta con la que se encontraron los parámetros de diseño de los controladores; así como, la cantidad de casos que se utilizarán para diseñar las diferentes estrategias de control en el próximo capítulo. Se ha demostrado continuamente durante este estudio que Matlab y sus complementos, son herramientas muy poderosas y útiles para el desarrollo de sistemas de control, tales como el control de cruceo adaptativo y los sistemas de ayuda a la conducción que existen actualmente en los vehículos comerciales. En este capítulo se han sentado las bases para poder llegar a los resultados que han sido analizados en el siguiente apartado.





# 8. Resultados

En esta sección se describen los resultados obtenidos tras aplicar la metodología de la sección 7, en conjunto con la utilización del banco de pruebas de la sección 6 y la integración de los modelos propuestos en la sección 5.

## 8.1 Selección de Parámetros

Para llevar a cabo la selección de parámetros que sean adecuados para cada controlador se tomó como escenario inicial el número uno, del cual se concluyeron los valores presentados en los siguientes apartados para cada estrategia de control.

### 8.1.1 Clásico PID

Para diseñar el controlador PID, se utilizó el método de Ziegler-Nichols. Inicialmente, se establecieron las constantes integral y derivativa en cero, variando únicamente el valor de la constante proporcional hasta lograr hallar una respuesta oscilatoria con amplitud constante. Para esto, se modificó el valor de la constante proporcional. El primer valor donde se presentó una oscilación con amplitud constante fue con una  $K_u=1$ . Utilizando el valor anterior y mediante el uso de la Tabla 1 se obtienen los valores siguientes,  $K_p=0.6$ ,  $K_i=0.1428$  y  $K_d=0.63$ .

Controlador	$K_p$	$\tau_i$	$\tau_d$
P	$0.5 * K_u$	$\infty$	0
PI	$0.45 * K_u$	$\frac{1}{2} * P_u$	0
PID	$0.6 * K_u$	$0.5 * P_u$	$0.125 * P_u$

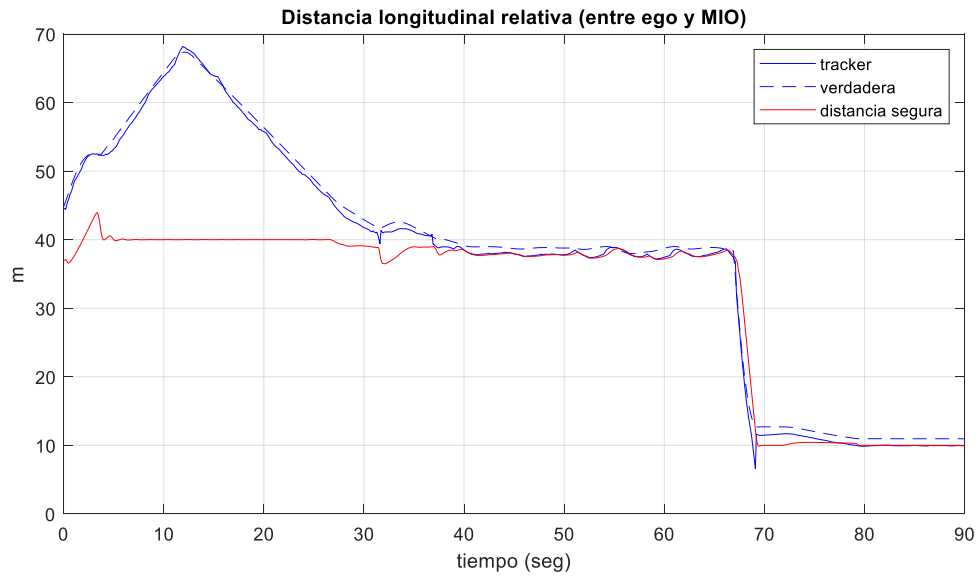
Tabla 8.1.1.1. método Ziegler-Nichols para sistemas cerrados

De lo anterior se obtuvieron los parámetros mostrados en la Tabla 8.1.1.2, así como el comportamiento del controlador PID, mostrado en la Figura 8.1.1.1.

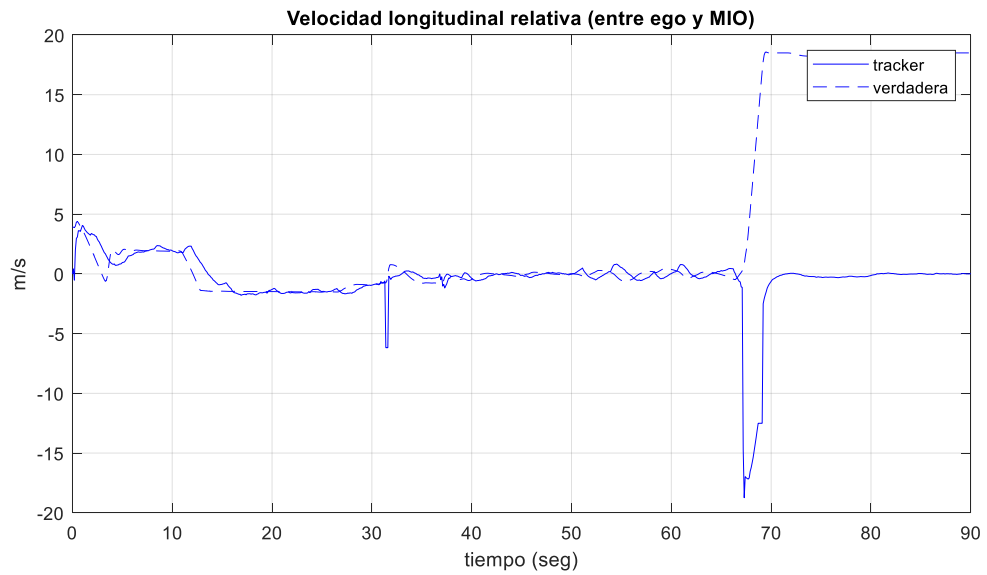
Parámetro	Símbolo	Valor
Tiempo de muestreo	$T_s$	0.1 s
Retraso	$T$	0.5 s
Constante proporcional	$K_p$	0.6
Constante integral	$k_i$	0.1428
Constante derivativa	$k_d$	0.63
Límite de aceleración superior	$A_{max}$	2 m/s <sup>2</sup>
Límite de aceleración inferior	$A_{min}$	-3 m/s <sup>2</sup>

Tabla 8.1.1.2. Parámetros del controlador PID

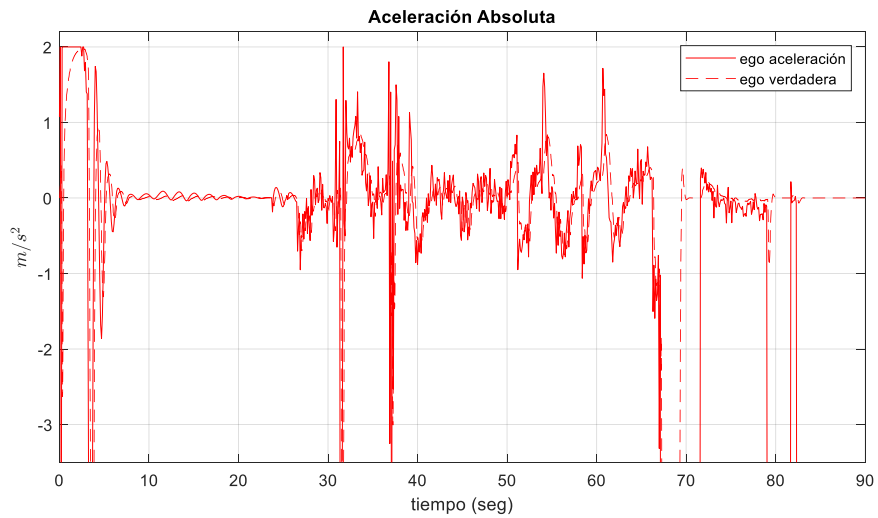
a)



b)



c)



d)

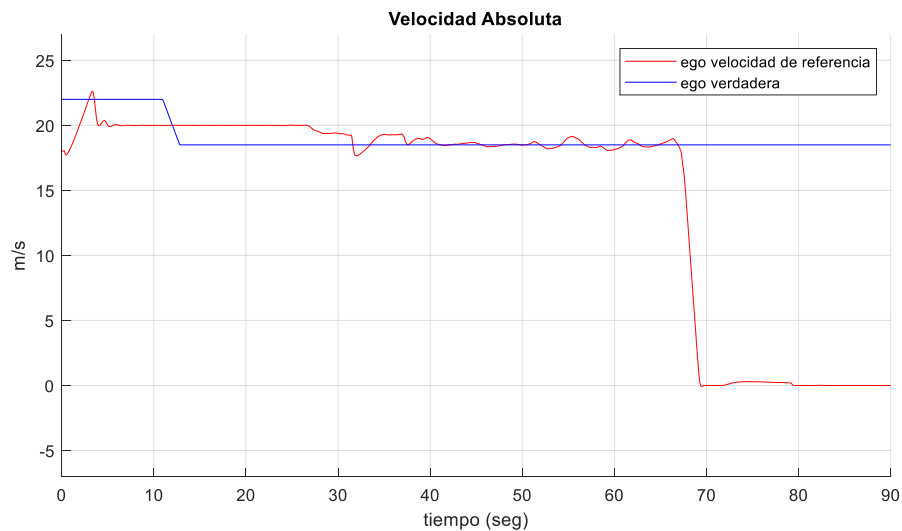


Figura 8.1.1.1. Desempeño del controlador PID de ACC, a) Distancia longitudinal relativa, b) velocidad longitudinal relativa, c) Aceleración absoluta, d) Velocidad absoluta

- **a) Distancia longitudinal relativa:** de esta gráfica se obtienen los siguientes valores de distancia: real 10.98 [m], tracker 10 [m]  $\pm$  5 [cm], deseada 10 [m].
- **b) Velocidad longitudinal relativa:** se observa que el controlador funciona, ya que reduce la diferencia de la velocidad entre ambos vehículos.

- **c) Aceleración absoluta:** En el caso particular de este controlador, muestra un comportamiento bastante bueno ante este escenario, ya que varía la aceleración de manera suave y en valores pequeños. Lo anterior es lo que se busca al controlar la aceleración del vehículo.
- **d) Velocidad absoluta:** En este caso se observa que el controlador realiza un cambio de velocidad bastante suave cuando se presenta una perturbación; además, mantiene un error bajo y prácticamente constante durante todo el tiempo de simulación restante hasta que el vehículo realiza el frenado de emergencia.

### 8.1.2 Constant-Time Gap Law (CTG)

Para diseñar el controlador por tiempo de espaciado constante (CTG por sus siglas en inglés) se modificó el factor lambda  $\lambda_{ctg}$  en valores de 0.1 a 0.5, de la Figura 8.1.2.1 podemos observar que el valor óptimo de este factor se encuentra entre 0.1 y 0.2, ya que se presenta una menor oscilación ante una perturbación al sistema; sin embargo, cuando el factor lambda es  $\lambda_{ctg} = 0.2$  presenta un mejor comportamiento cualitativo, ya que sigue la referencia inicial y el cambio de velocidad de una manera más eficaz. Lo anterior se verifica cuando se compara el error de cada uno de estos valores, tal y cómo se muestra en la Figura 8.1.2.2, resultando así un error RMSE en la velocidad longitudinal de 0.2295.

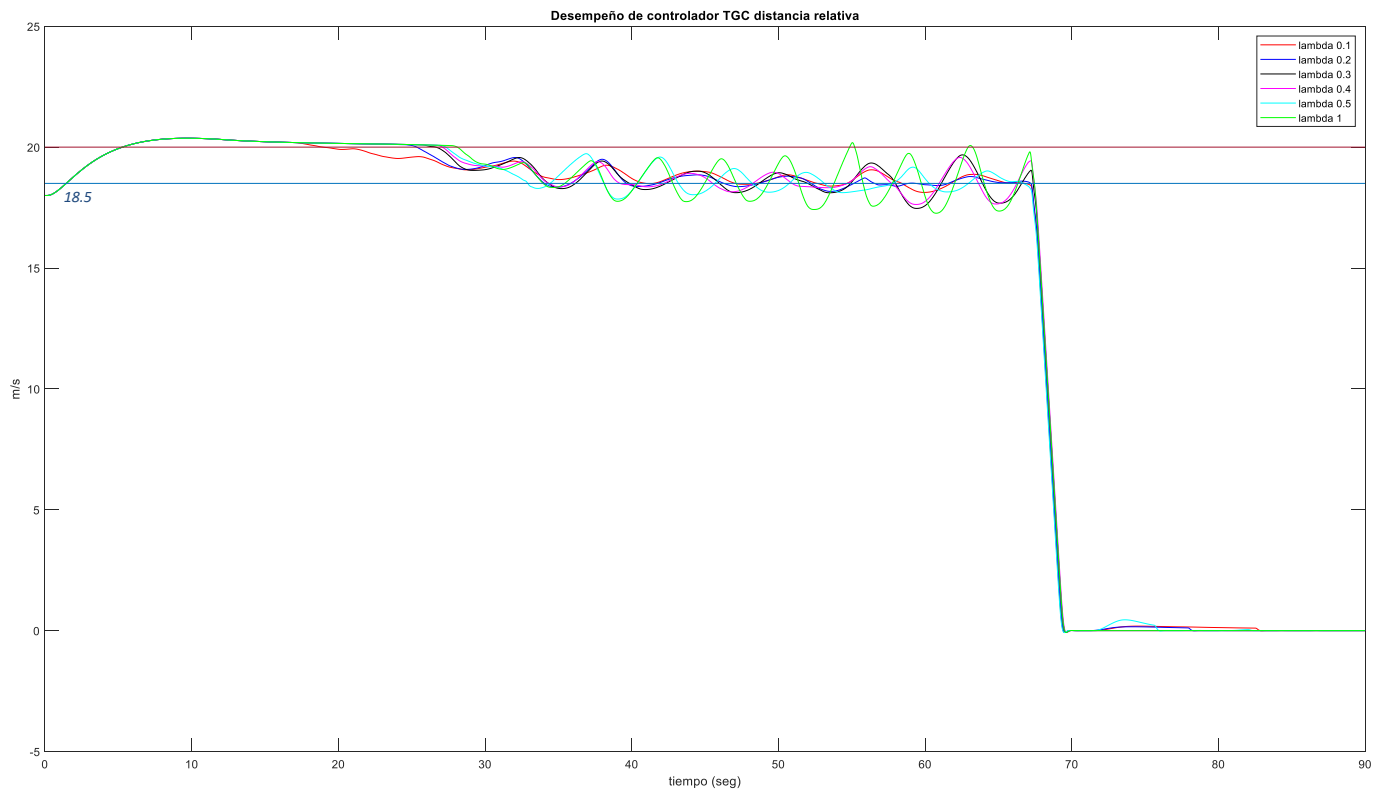


Figura 8.1.2.1. Variación de la velocidad longitudinal con respecto a lambda

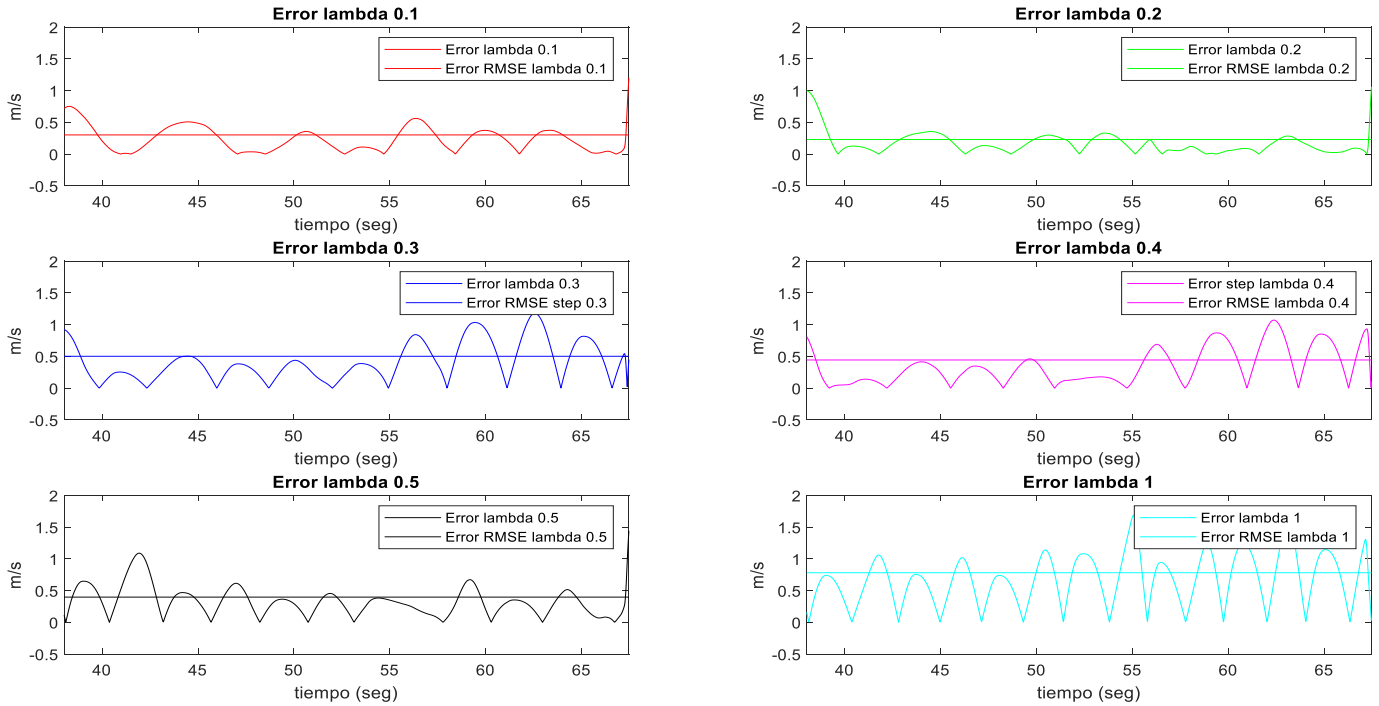


Figura 8.1.2.2. Variación del error RMSE al variar lambda

Continuando con el diseño, se modificó el tiempo de avance (*headway time*  $h$ ) en valores de 0.5, 1 y 1.5 segundos. Variar este parámetro en el controlador CTG resultó afectar la estabilidad de nuestro controlador de manera considerable, tal y como se observa en la Figura 8.1.2.3. Se puede señalar que, al utilizar valores pequeños de  $h$ , la amplitud de la señal comienza a engrandecerse y resulta en una divergencia con respecto al valor deseado; por otro lado, valores de 1 [s] y 1.5 [s] muestran una mejoría en el comportamiento, siendo el segundo con el cual se consigue una menor amplitud y que converja al valor deseado con mayor rapidez.

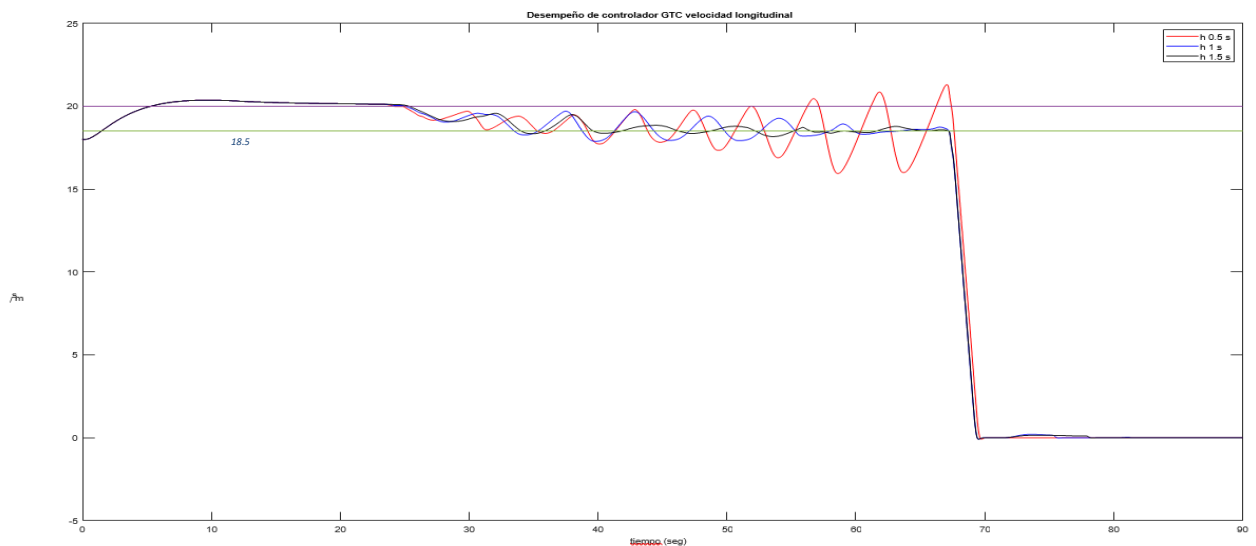


Figura 8.1.2.3. Variación de la velocidad longitudinal con respecto al cambio de  $h$  (*headway time*).

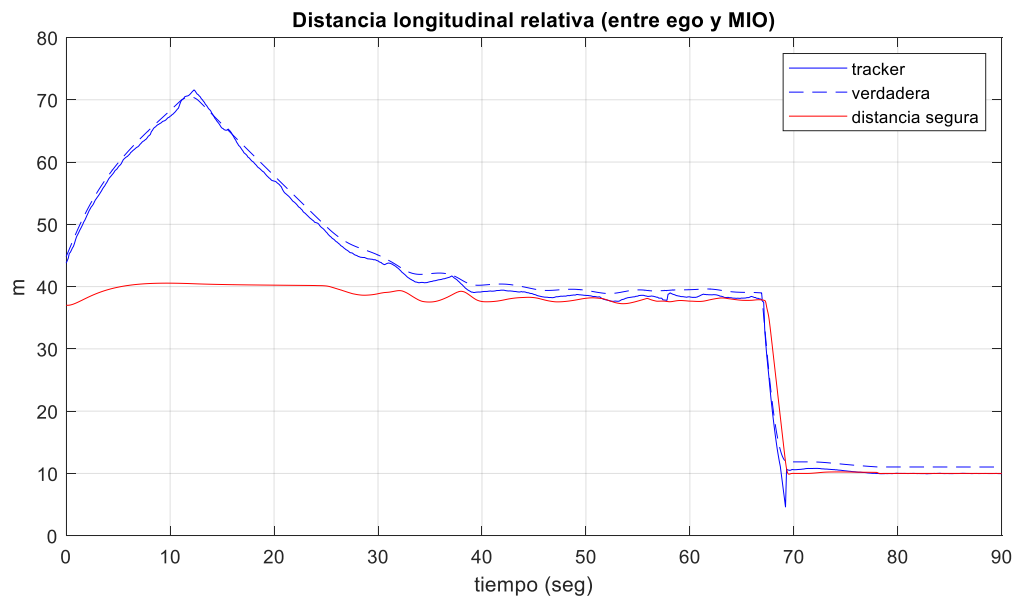
Por tal motivo, los parámetros seleccionados para este controlador se muestran en la Tabla 8.1.2.

Parámetro	Símbolo	Valor
Tiempo de muestreo	$T_s$	0.1 s
Retraso	$\tau$	0.5 s
Tiempo de avance (headway time)	$h$	1.5 s
Factor lambda	$\lambda_{ctg}$	0.2
Límite de aceleración superior	$A_{max}$	2 m/s <sup>2</sup>
Límite de aceleración inferior	$A_{min}$	-3 m/s <sup>2</sup>

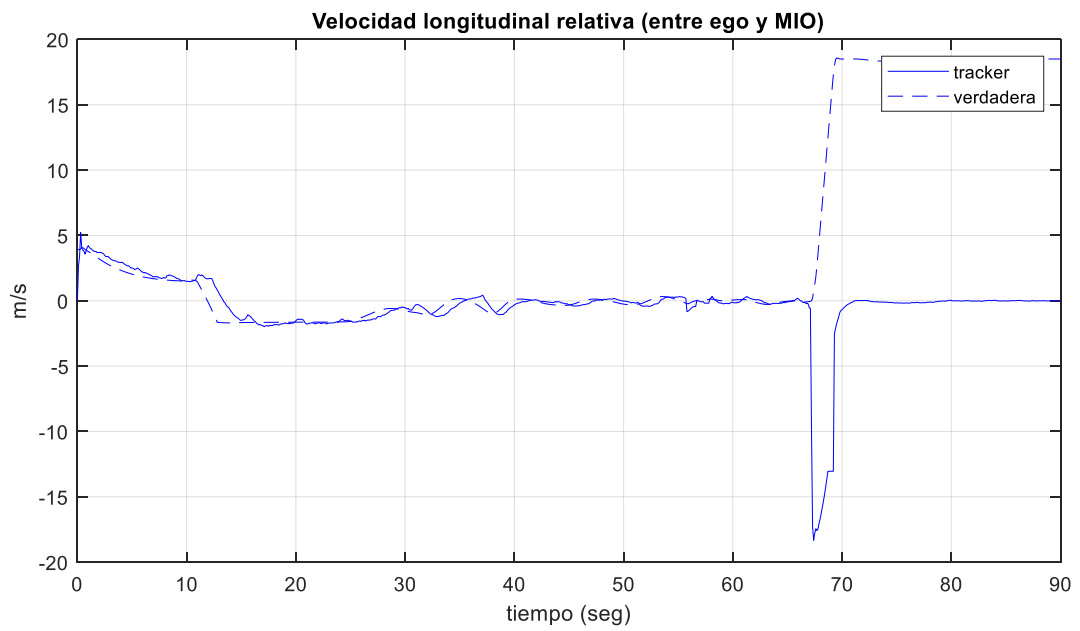
Tabla 8.1.2. Parámetros del controlador

A continuación, se presenta en la Figura 8.1.2.4 el desempeño general del controlador CTG de cruceo adaptativo.

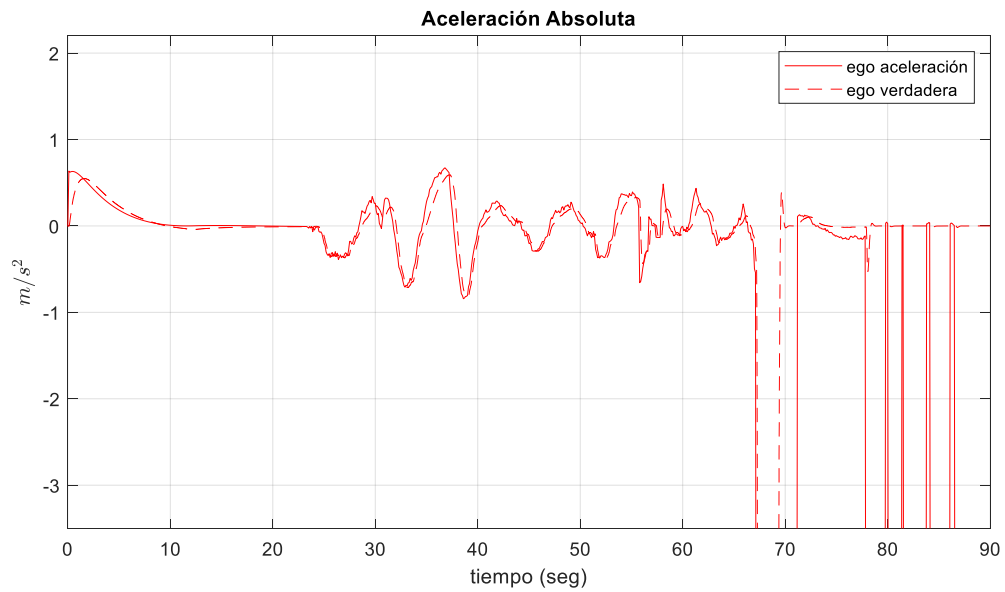
a)



b)



c)





d)

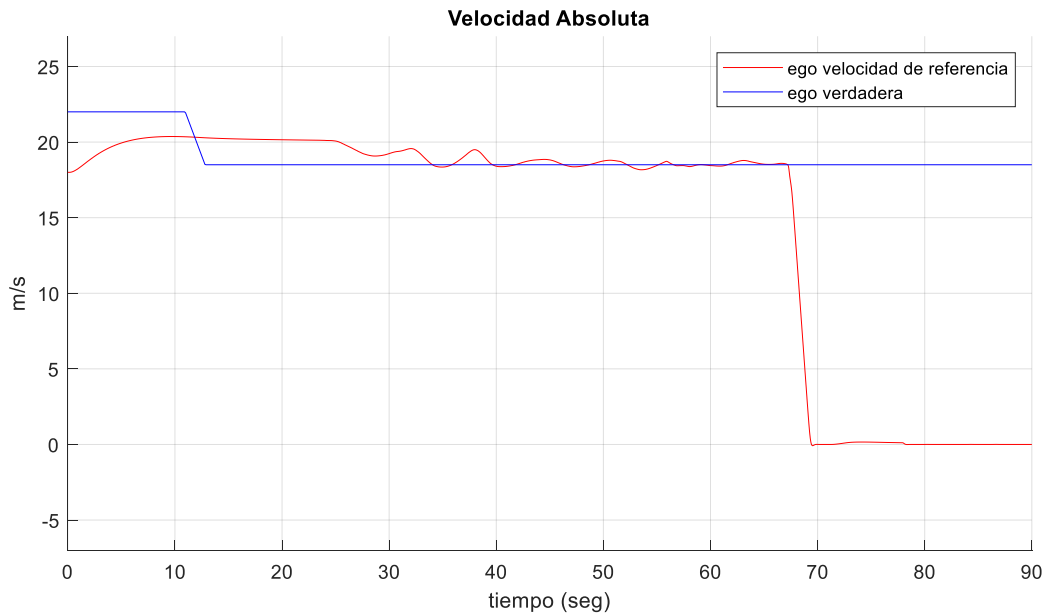


Figura 8.1.2.4. Desempeño del controlador CTG de ACC, a) Distancia longitudinal relativa, b) velocidad longitudinal relativa, c) Aceleración absoluta, d) Velocidad absoluta

- **a) Distancia longitudinal relativa:** de esta gráfica se obtienen los siguientes valores de distancia: real 11.03 [m], *tracker* 10 [m]  $\pm$  5 [cm], deseada 10 [m]. De lo anterior, es relevante que la distancia calculada mediante el algoritmo de visión es prácticamente el valor deseado, con tan solo una variación de algunos centímetros.
- **b) Velocidad longitudinal relativa:** se observa que el controlador funciona al reducir la diferencia de la velocidad entre ambos vehículos.
- **c) Aceleración absoluta:** Para este controlador se muestra una tendencia de cambio con valores más grandes al inicio de la perturbación, pero conforme avanza el vehículo se observa que disminuyen la amplitud de la señal de aceleración y desaceleración.
- **d) Velocidad absoluta:** Al igual que la aceleración, para este escenario la velocidad tiende a converger en el valor de referencia antes de frenar por completo.

### 8.1.3 Sliding mode controller

Primeramente, se modificó el parámetro eta ( $\eta$ ) el cual modifica la llamada condición de llegada (“*reaching condition*” en inglés), los resultados obtenidos se muestran en la Figura 8.1.3.1, en donde se modificó el valor de  $\eta$  de 1 hasta 10 en pasos de 1.

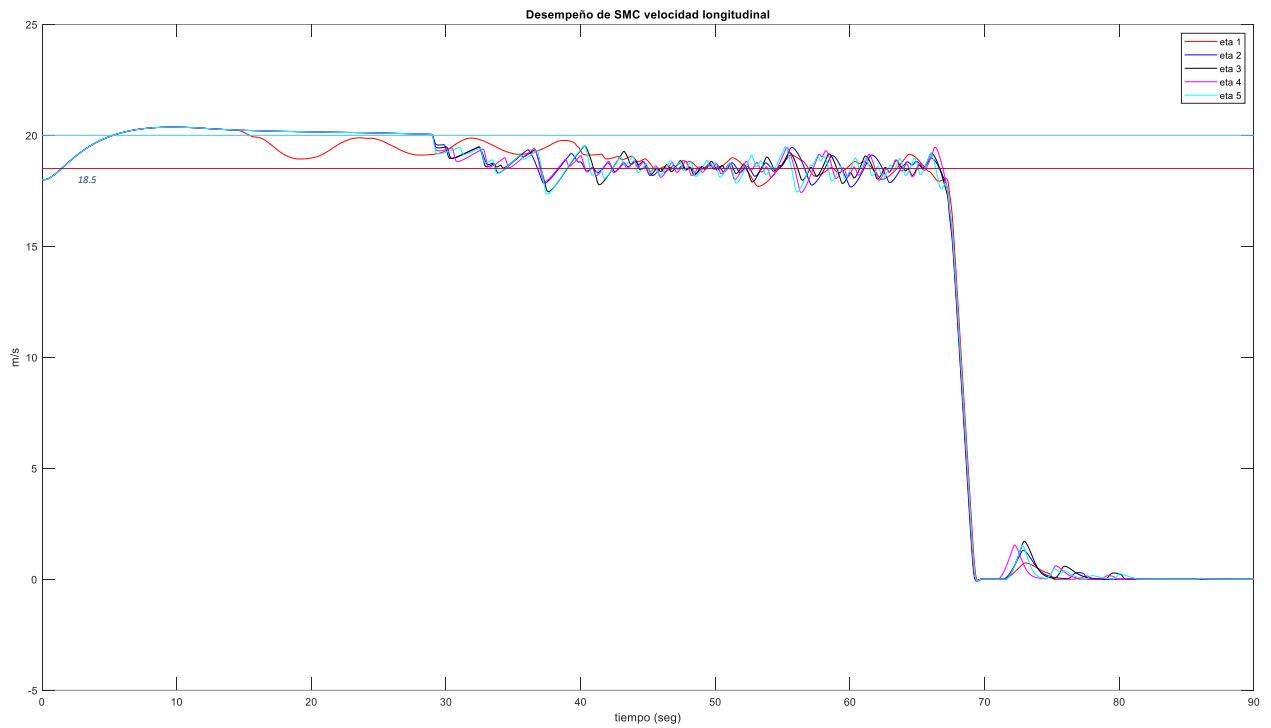


Figura 8.1.3.1. Efecto de la variación de eta (1-5) en la velocidad longitudinal del vehículo

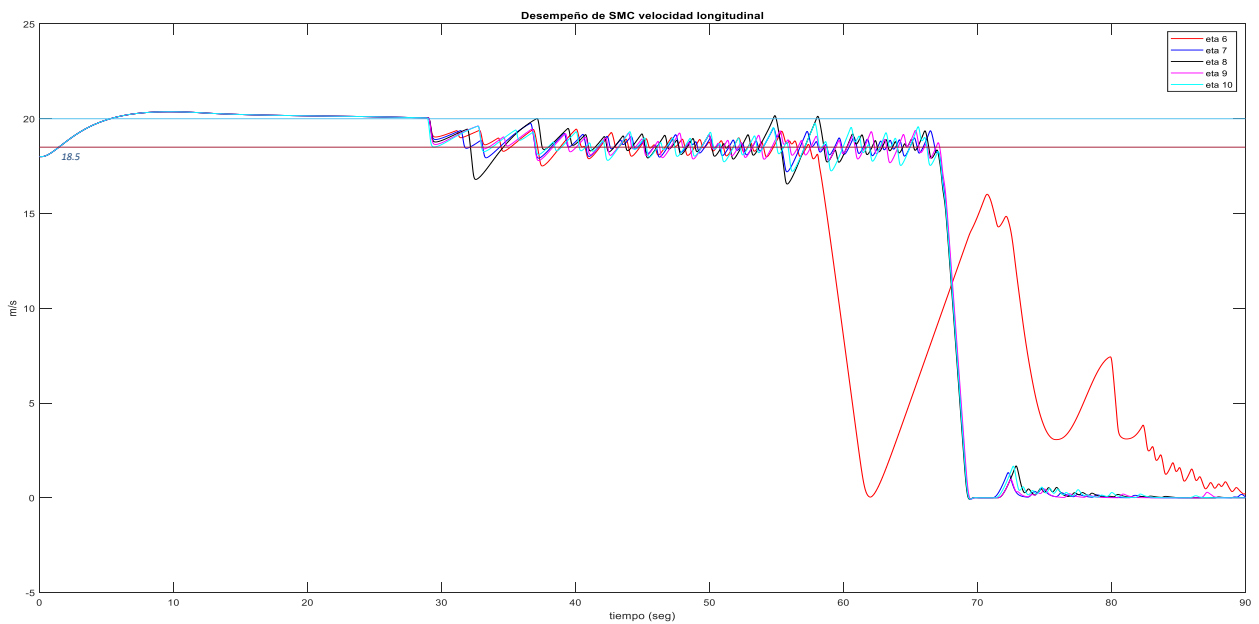


Figura 8.1.3.2. Efecto de la variación de eta (6-10) en la velocidad longitudinal del vehículo

Posteriormente se calculó el error absoluto promedio de cada uno de los valores, obteniéndose así la Tabla 8.1.3.1 y las Figuras 8.1.3.3-8.1.3.4.

Eta	Error
1	0.4596
2	0.3312
3	0.3489
4	0.3306
5	0.3659
6	0.4094
7	0.3727
8	0.5688
9	0.3661
10	0.4899

Tabla 8.1.3.1. Error absoluto promedio al variar eta

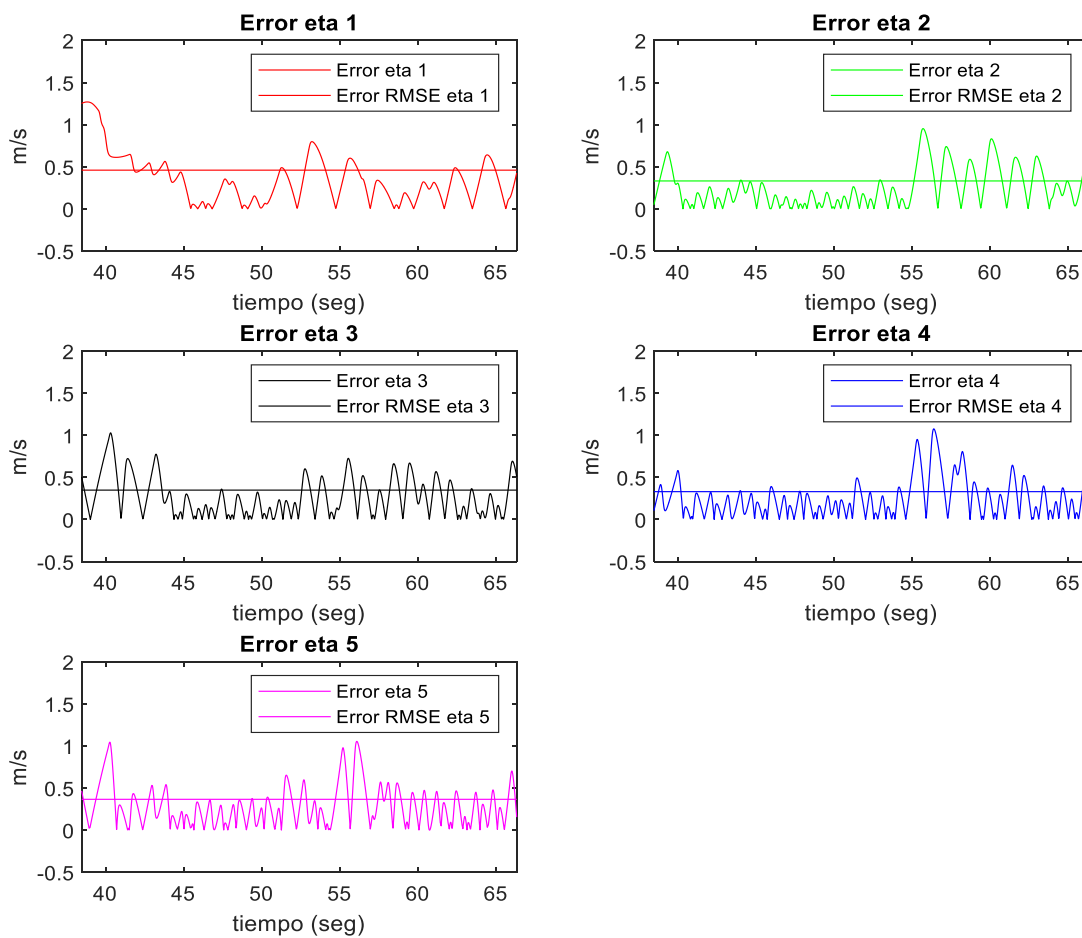


Figura 8.1.3.3. Variación del error absoluto al variar eta de 1 a 5

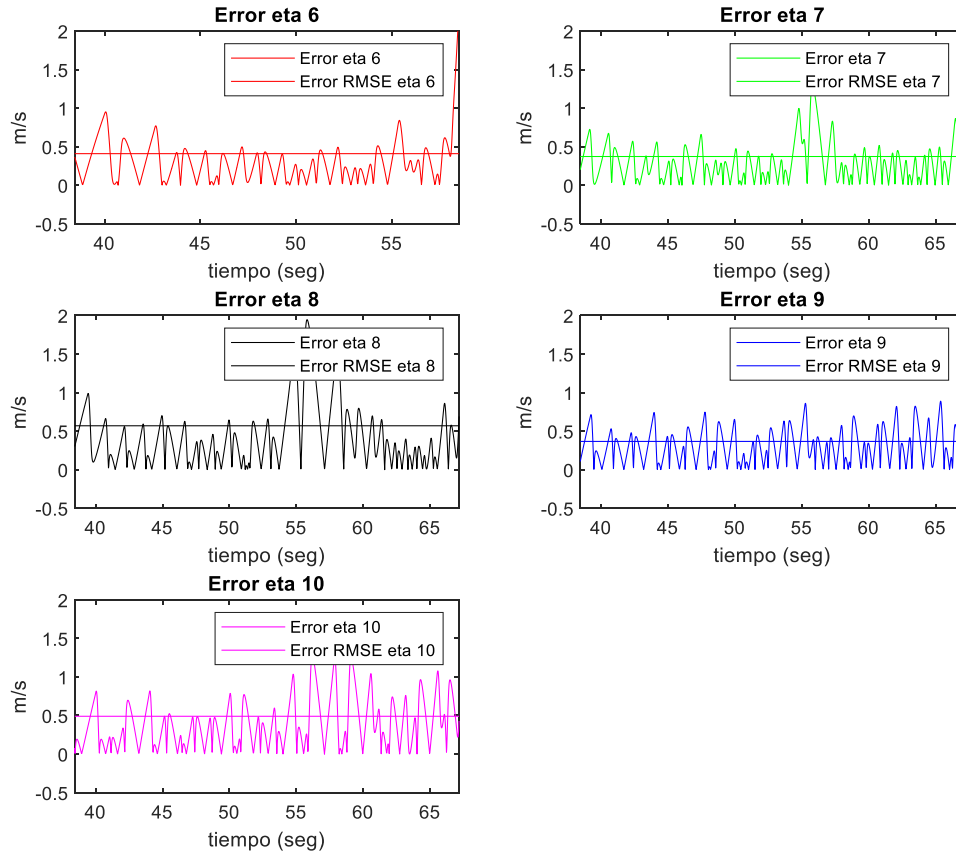


Figura 8.1.3.4. Variación del error absoluto al variar eta de 6 a 10

De las Figuras 8.1.3.3 y 8.1.3.4 se observa que el valor con un mayor ajuste a la referencia obteniendo un menor error comparado con los valores restantes, es cuando se utiliza un valor de  $\eta=4$ . En la mayoría de los valores se muestra cierta amplitud constante en donde intenta disminuir el error a cero; sin embargo, podemos visualizar que mientras más grande es el valor de  $\eta$ , la amplitud incrementa considerablemente. Por otro lado, mientras menor sea el valor de  $\eta$ , la amplitud decrece hasta cierto punto.

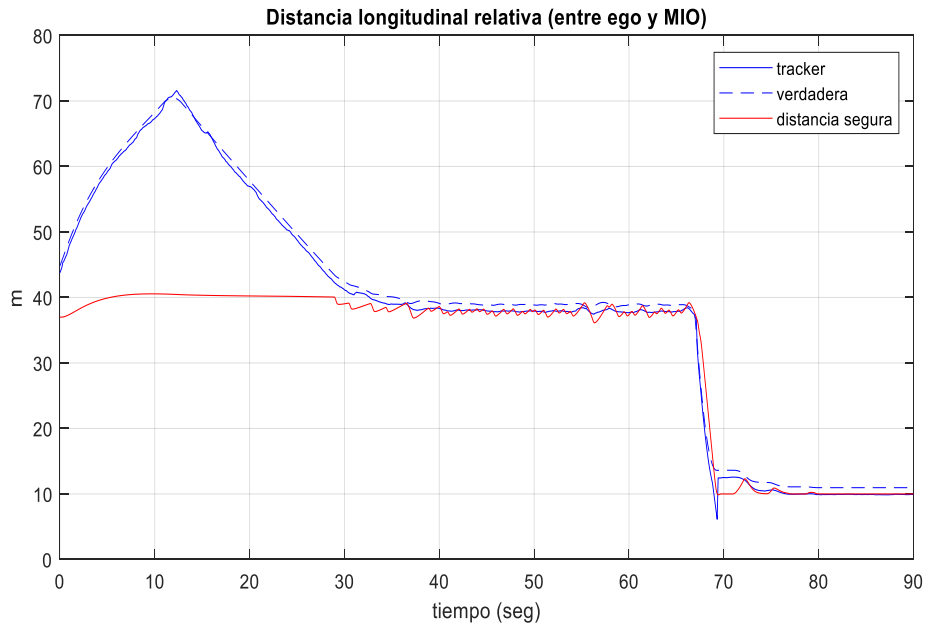
Por lo cual los parámetros seleccionados para el controlador SMC se muestran en la Tabla 8.1.3.2.

Parámetro	Símbolo	Valor
Tiempo de muestreo	$T_s$	0.1 s
Retraso	$\tau$	0.5 s
Tiempo de avance (headway time)	$h$	1.5 s
Factor eta	$\eta$	4
Límite de aceleración superior	$A_{max}$	2 m/s <sup>2</sup>
Límite de aceleración inferior	$A_{min}$	-3 m/s <sup>2</sup>

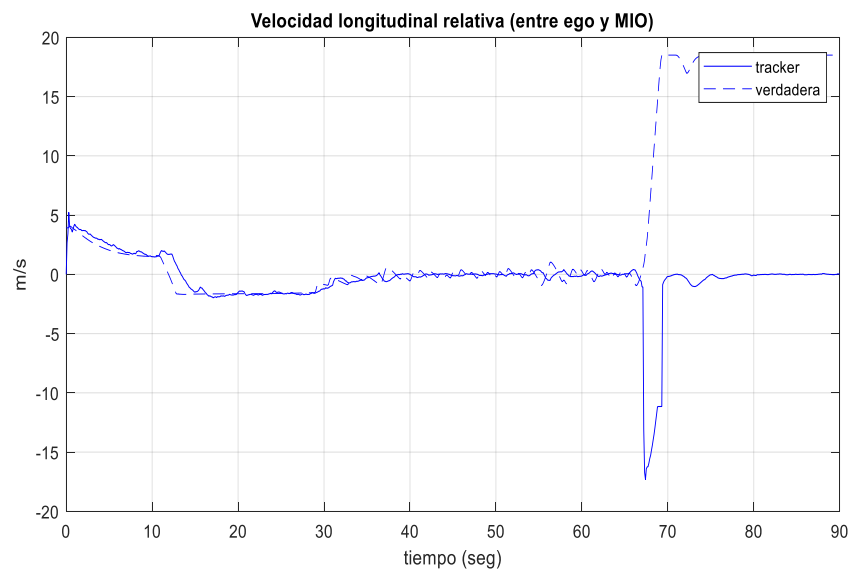
Tabla 8.1.3.2. Parámetros del controlador SMC

A continuación, se presenta en la Figura 8.1.3.5 el desempeño general del controlador SMC de cruceo adaptativo.

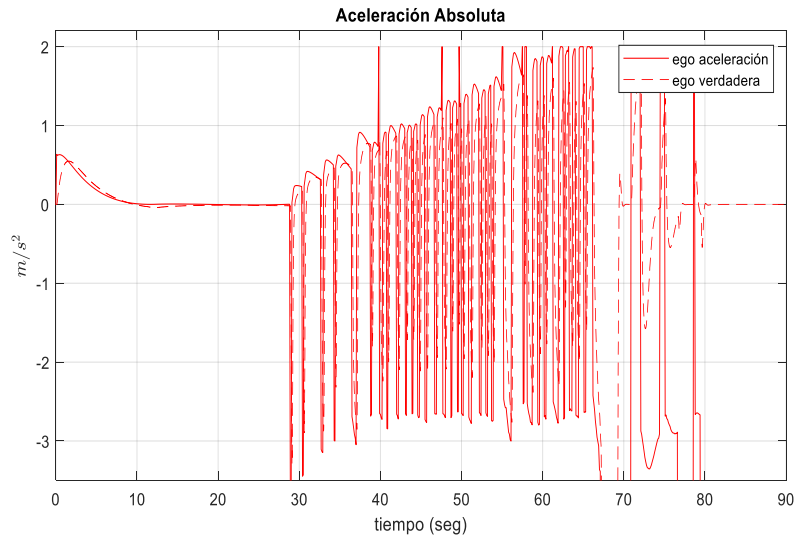
a)



b)



c)



d)

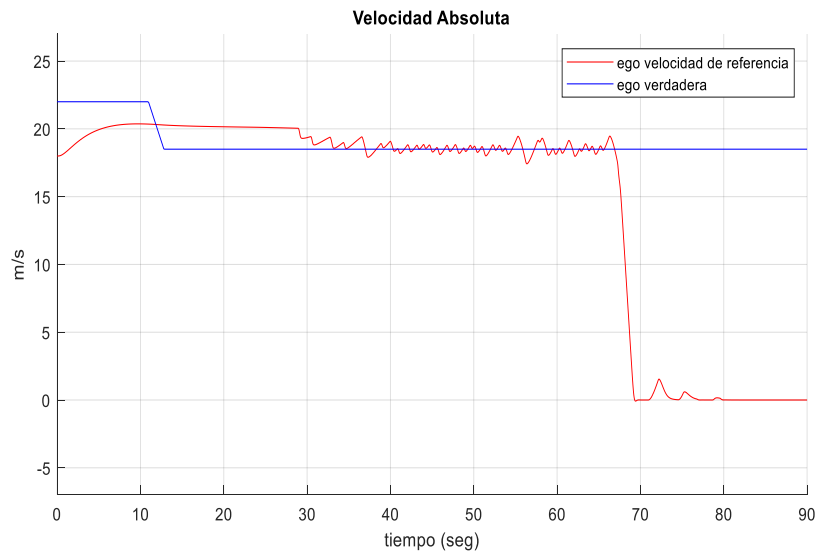


Figura 8.1.3.5. Desempeño del controlador SMC de ACC, a) Distancia longitudinal relativa, b) velocidad longitudinal relativa, c) Aceleración absoluta, d) Velocidad absoluta

- **a) Distancia longitudinal relativa:** de esta gráfica se obtienen los siguientes valores de distancia: real 10.95 [m], *tracker* 10 [m]  $\pm$  12 [cm], deseada 10 [m].

- **b) Velocidad longitudinal relativa:** se observa que el controlador funciona, ya que reduce la diferencia de la velocidad entre ambos vehículos.
- **c) Aceleración absoluta:** Para este controlador se muestra una tendencia de cambio influenciado por el efecto de parloteo o mejor conocido como “*chattering*”, ya que los valores de la señal de control, en este caso la aceleración, cambian abruptamente entre el límite máximo y mínimo de aceleración, este tipo de comportamiento suele ser dañino para los elementos mecánicos, por lo cual es un comportamiento tentativamente no deseado entre nuestros controladores.
- **d) Velocidad absoluta:** Al igual que la aceleración, para este escenario la velocidad tiende a converger en el valor de referencia antes de frenar por completo; sin embargo, presenta cierta perturbación entre el periodo de tiempo comprendido de 50-60 [s] de la simulación, en donde se observa como vuelve a incrementar la amplitud de la señal, pero logra recuperar su comportamiento anterior y seguir el valor de referencia.

### 8.1.4 MPC Model Predictive Control

Para el controlador MPC se definió un modelo mediante espacio de estados que describe un vehículo de dos ruedas, que contiene y describe el movimiento lateral y longitudinal del mismo.

Posteriormente, se comparó el comportamiento cuando se modifica la robustez (R) del controlador, se incrementó este valor en valores constantes desde 0.1 hasta 1.0. De lo anterior, se

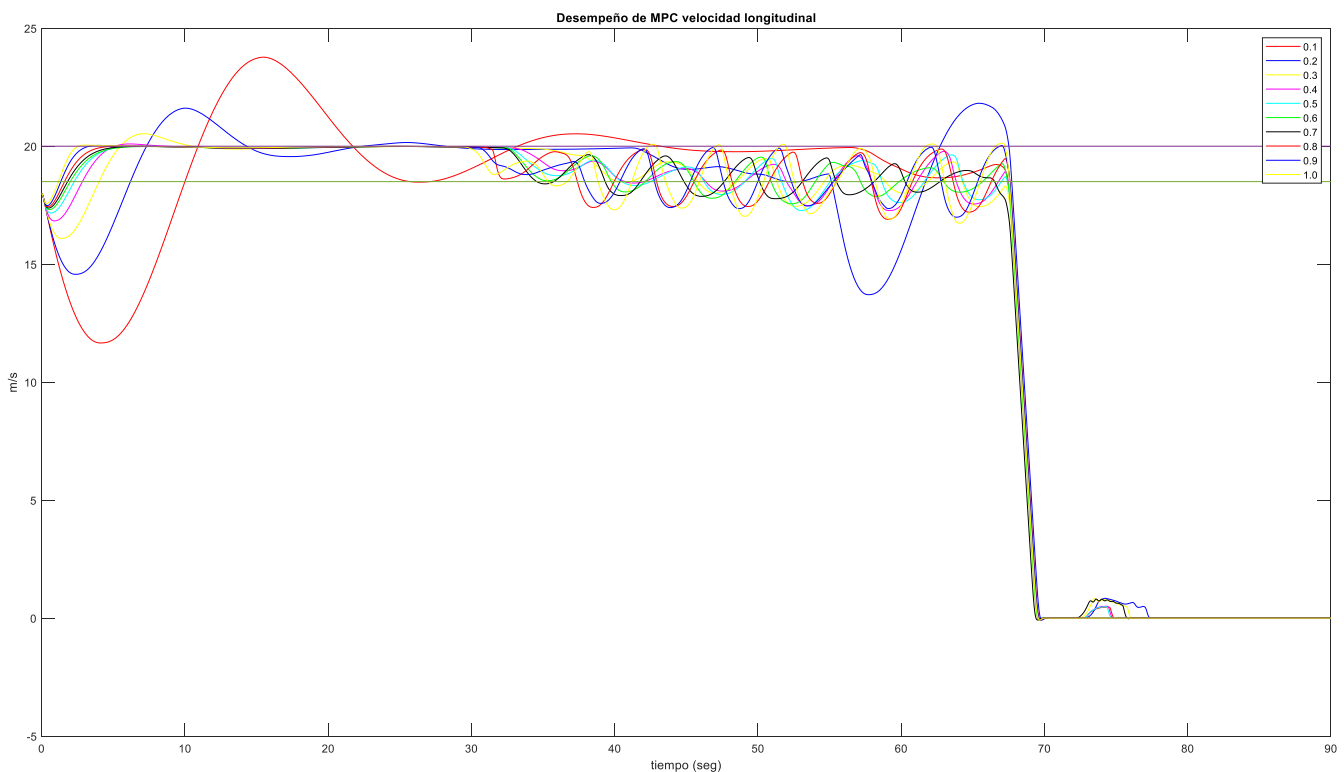


Figura 8.1.4.1. Variación de la robustez del controlador MPC, comportamiento de la velocidad longitudinal

obtuvo la Figura 8.1.4.1 en donde se observaron principalmente dos cosas, el tiempo que tarda nuestra señal en llegar a los valores de referencia, así como los valores de sobrepaso y oscilación a lo largo del experimento. Se pueden identificar dos partes principales, la primera de ellas es el comportamiento del controlador de cruce, en donde mientras más incrementa el valor, este tiende a llegar el valor de referencia con mayor rapidez; la segunda, en donde se presenta la perturbación (cambio de velocidad del vehículo MIO de 22 m/s a 18m/s), la amplitud de las oscilaciones disminuye; sin embargo, estas no desaparecen del todo, tal y cómo se observa en la Figura 8.1.4.1.

De manera similar, se puede observar en la Figura 8.1.4.2 la variación que presenta la distancia relativa entre el vehículo EGO y el MIO correspondiente a cada valor de robustez. Se sigue de la misma manera, que los valores cercanos a 0.4, 0.5, 0.6 son aquellos que presentan un mejor desempeño tratando de alcanzar los valores de referencia, además de una buena distancia de separación al momento de frenar súbitamente, la cual se modifica acorde al cada valor

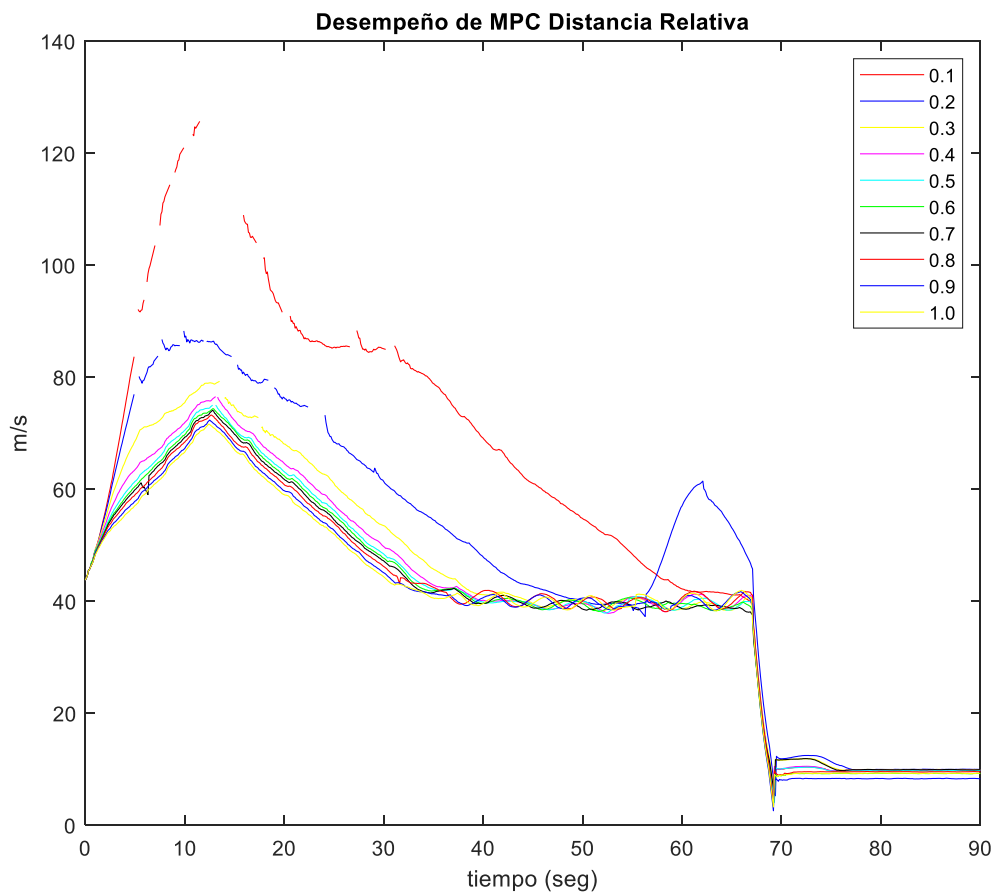


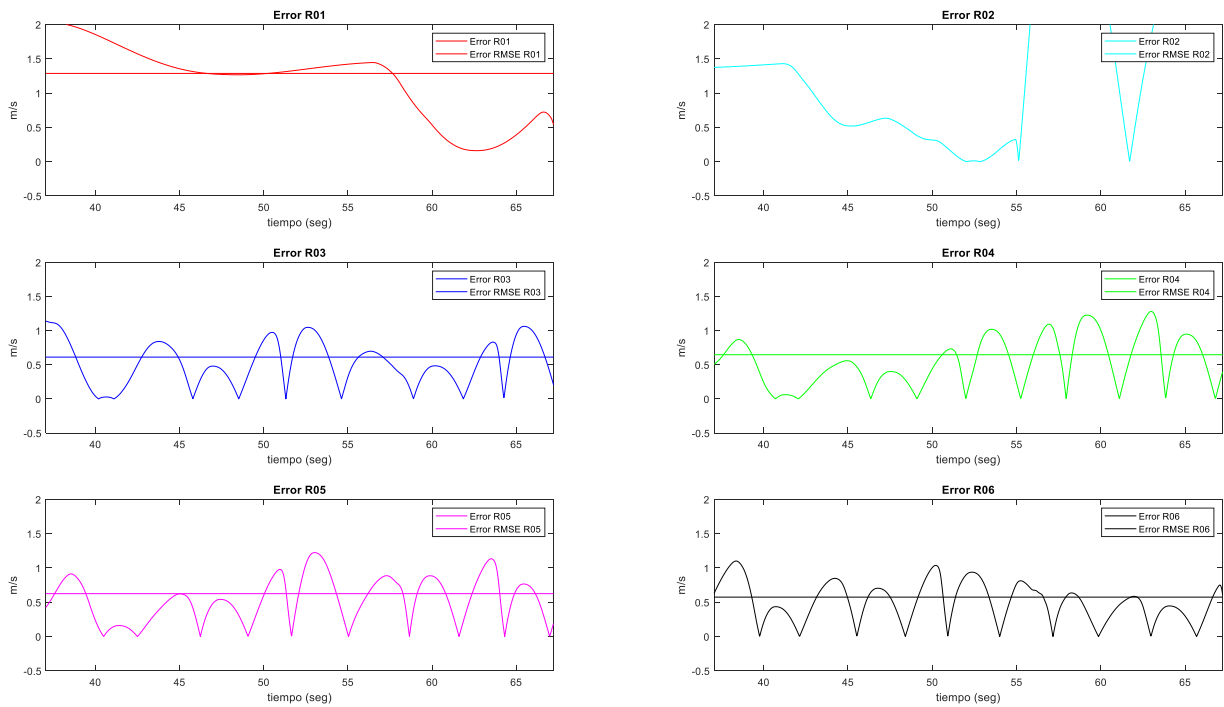
Figura 8.1.4.1. Variación de la robustez del controlador MPC, comportamiento de la distancia relativa entre EGO y MIO



En la Figura 8.1.4.1 se presentan resultados cualitativos de este primer escenario de simulación; sin embargo, complementariamente se realizó una comparación de manera cuantitativa, en la cual se calculó el error mediante el uso de mínimos cuadrados (RMSE) de cada una de las variantes con respecto a la referencia, es decir, cuando el vehículo EGO cambia su velocidad a 18.5 [m/s] y hasta antes de frenar completamente durante el intervalo de tiempo comprendido entre  $t$  [36,68] segundos. De lo anterior se obtuvo la Tabla 8.1.4.1 y Figura 8.1.4.3.

Robustez (R)	RMSE
<b>0.1</b>	1.2869
<b>0.2</b>	2.0935
<b>0.3</b>	0.6120
<b>0.4</b>	0.6470
<b>0.5</b>	0.6256
<b>0.6</b>	0.5751
<b>0.7</b>	0.5652
<b>0.8</b>	0.9022
<b>0.9</b>	0.8958
<b>1.0</b>	1.05788

Tabla 8.1.4.1. Error RMSE de la velocidad longitudinal al variar la robustez



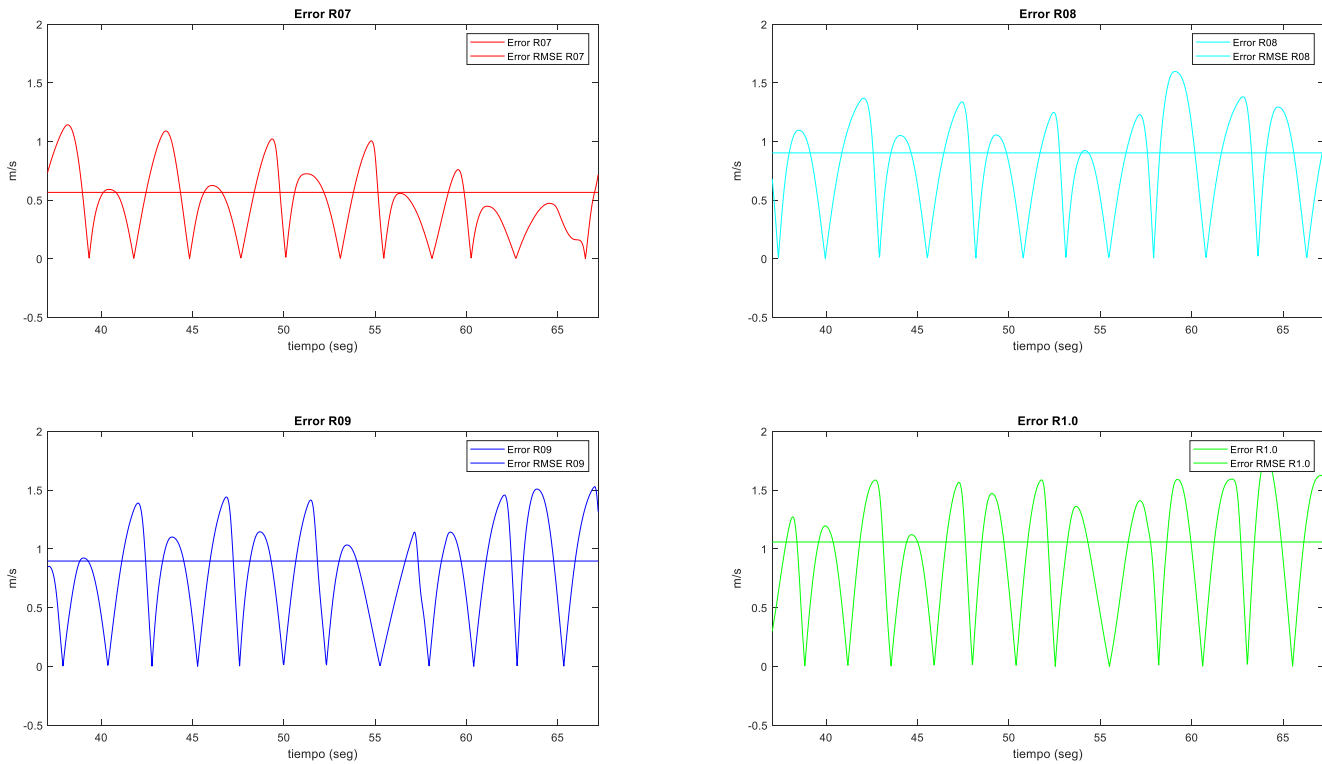


Figura 8.1.4.3. Error RMSE, Robustez: R1 0.1, R2 0.2, R3 0.3, R4 0.4, R5 0.5, R6 0.6, R7 0.7, R8 0.8, R9 0.9, R10 1.0

En un primer acercamiento, el valor óptimo de robustez para minimizar el error se puede encontrar entre 0.5 y 0.7, ya que son los valores donde el error es menor.

Continuando con el diseño, se modificó el horizonte de predicción ( $H_p$ ), para comprender el comportamiento del controlador. Para llevar a cabo este análisis, la robustez del controlador se mantuvo fija con un valor de 0.5 (valor neutro), inicialmente el horizonte de predicción se fijó en un valor de 30, posteriormente se modificó este valor en incrementos y decrementos de 10 en un rango de 20-50, agregando una última simulación con un valor de 100. Específicamente, el objetivo de modificar este parámetro es entender exactamente cómo varía el comportamiento del controlador cuando se modifica el horizonte de predicción, ya que teóricamente cuanto mayor sea este valor, el controlador realizará el cálculo de predicción en un intervalo de tiempo mayor del modelo propuesto, esto implica a su vez un uso de mayores recursos computacionales. En la Figura 8.1.4.4 y 8.1.4.5 se pueden observar los resultados obtenidos de la variación en la velocidad longitudinal y la distancia relativa entre los vehículos.

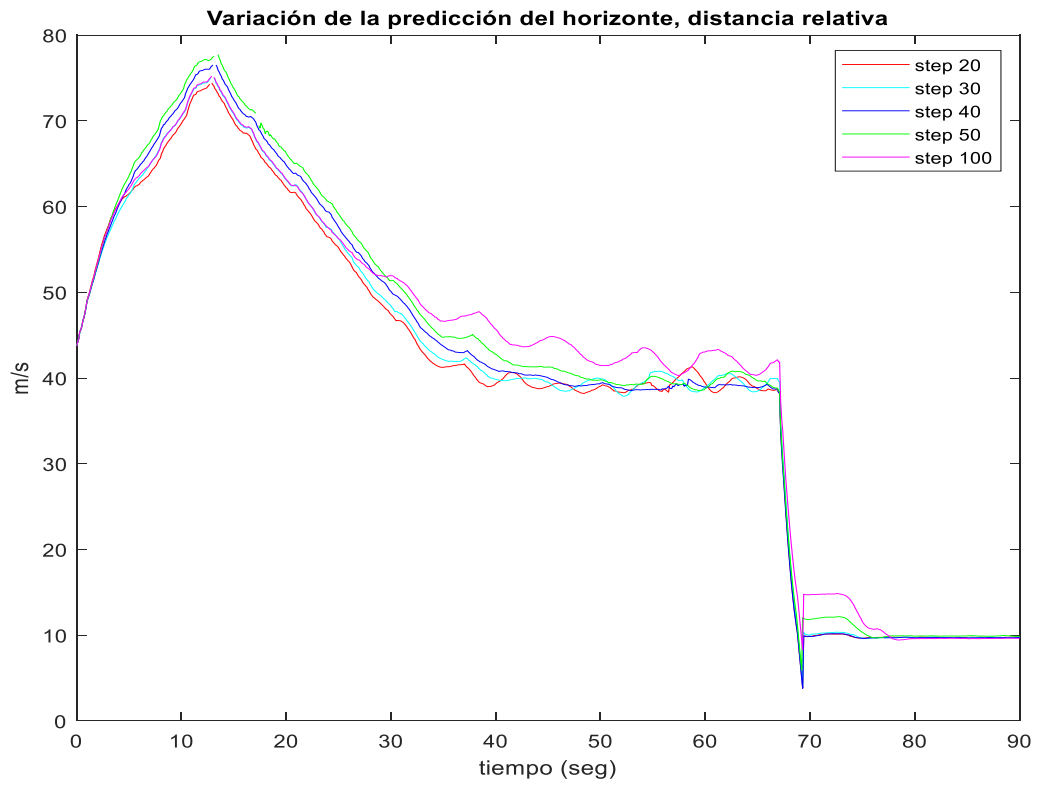


Figura 8.1.4. 4. Variación del horizonte de predicción del controlador MPC, comportamiento de la distancia relativa entre EGO y MIO

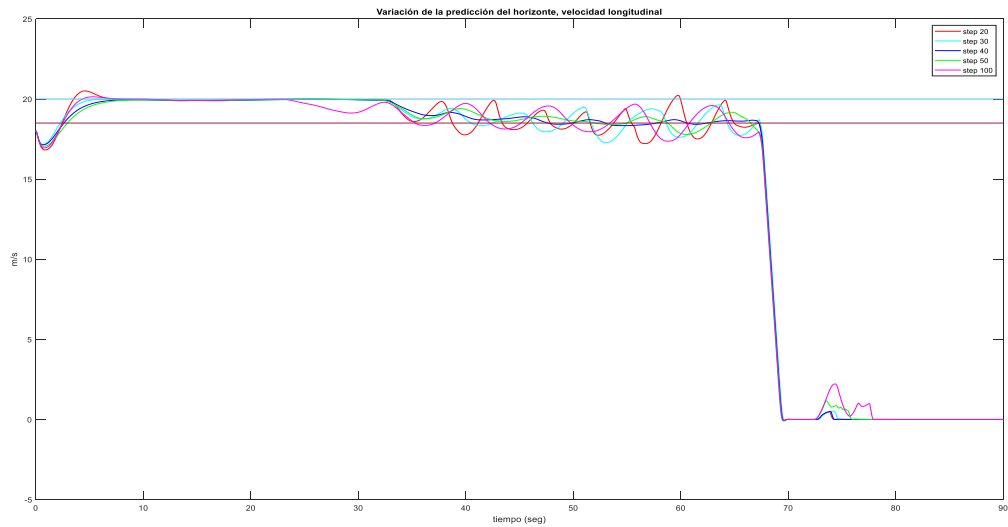


Figura 8.1.4.5. Variación del horizonte de predicción del controlador MPC, comportamiento de la velocidad longitudinal del EGO

Similar al análisis realizado con el parámetro de robustez, se comparó el error RMSE de cada uno de los horizontes de predicción, de donde se obtuvo la Tabla 8.1.4.2 y la Figura 8.1.4.6.

Horizonte de predicción	RMSE
20	0.7243
30	0.6249
40	0.2510
50	0.4056
100	0.7025

Tabla 8.1.4.2. Error RMSE de la velocidad longitudinal al variar el horizonte de predicción.

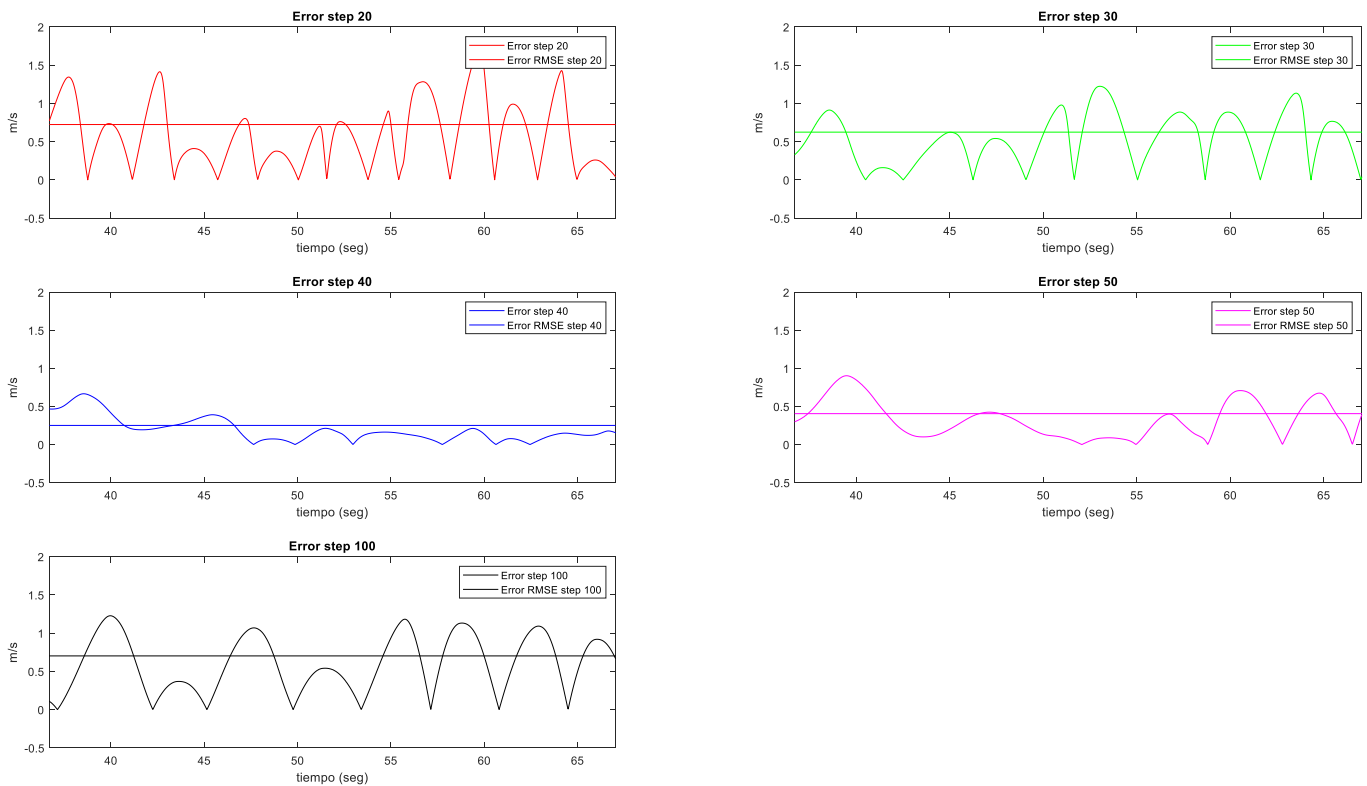


Figura 8.1.4.6. Error RMSE, Horizonte de predicción.

De la Figura 8.1.4.6 y la Tabla 8.1.4.2 podemos extraer que el valor más adecuado para el horizonte de predicción es 40, ya que con este valor el error RMSE es de 0.2510; no obstante, esto

puede estar sujeto al periodo de tiempo en el que se presente la perturbación, por lo que hasta este punto se establece este valor como el óptimo para realizar simulaciones posteriores.

Una vez que se obtuvieron los valores anteriores, tanto del horizonte de predicción (40) como de robustez (0.5-0.7), se combinaron para evaluar el desempeño del controlador, los resultados se observan en la Figura 8.1.4.7.

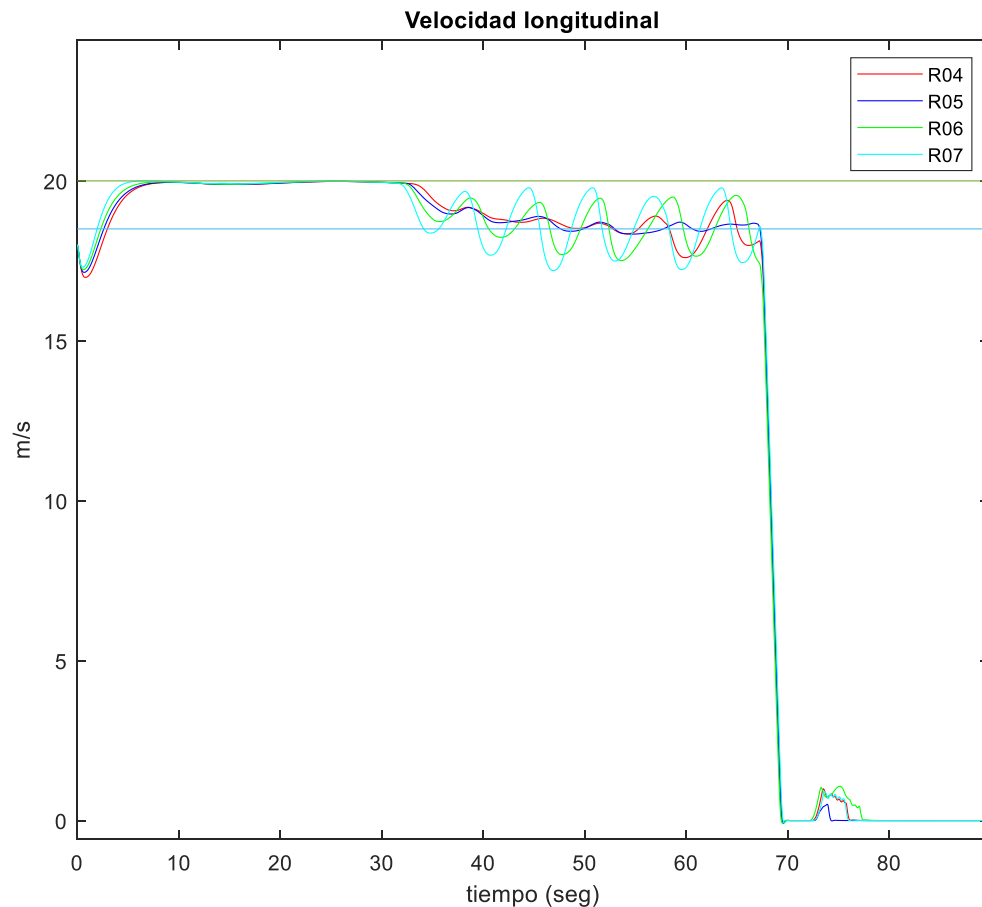


Figura 8.1.4.7. Variación de la robustez con un horizonte de predicción constante de 40

Como podemos observar de la Figura 8.1.4.7, los valores de simulación que favorecen el comportamiento son los siguientes: horizonte de predicción de 40 y robustez media de 0.5; sin embargo, aún queda por evaluar un parámetro más, el control de horizonte. Dicho parámetro se modificó en valores de 1 hasta 10, de igual forma se obtuvieron los resultados mostrados en la Figura 8.1.4.8.

Para modificar el parámetro control de horizonte, se utilizó la herramienta de diseño para MPC integrada en Matlab. Tal y como se observa en la Figura 8.1.4.8 hay ciertos valores que tienen un comportamiento estable y poco errático, dichos valores del control de horizonte son 1, 4, 5, 9.

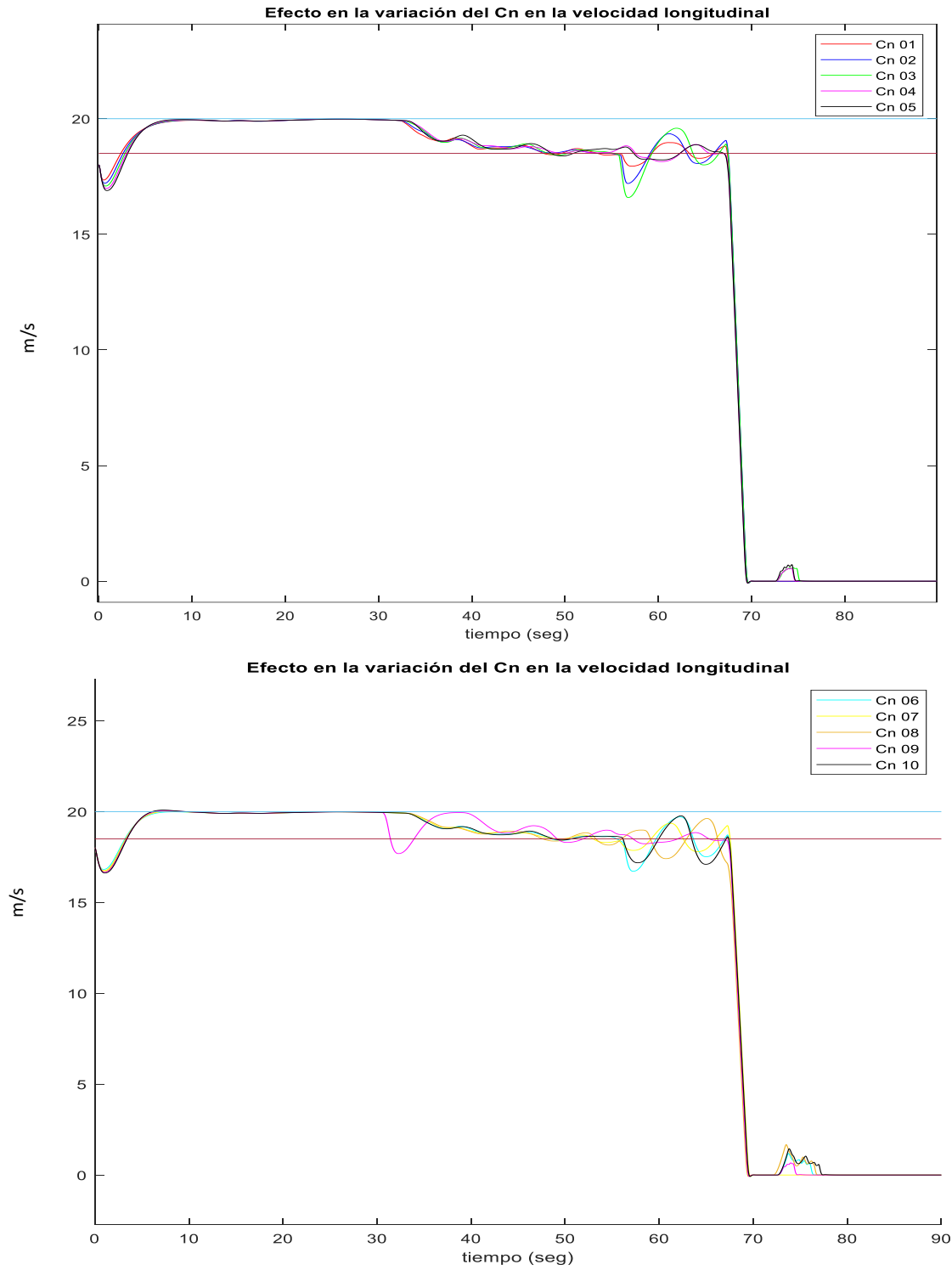


Figura 8.1.4.8. Efecto en la variación del  $N_c$  en la velocidad longitudinal

A continuación, se compararon los valores seleccionados y se calculó el error RMSE, de lo cual se obtiene la Figura 8.1.4.9 y la Tabla 8.1.4.3:

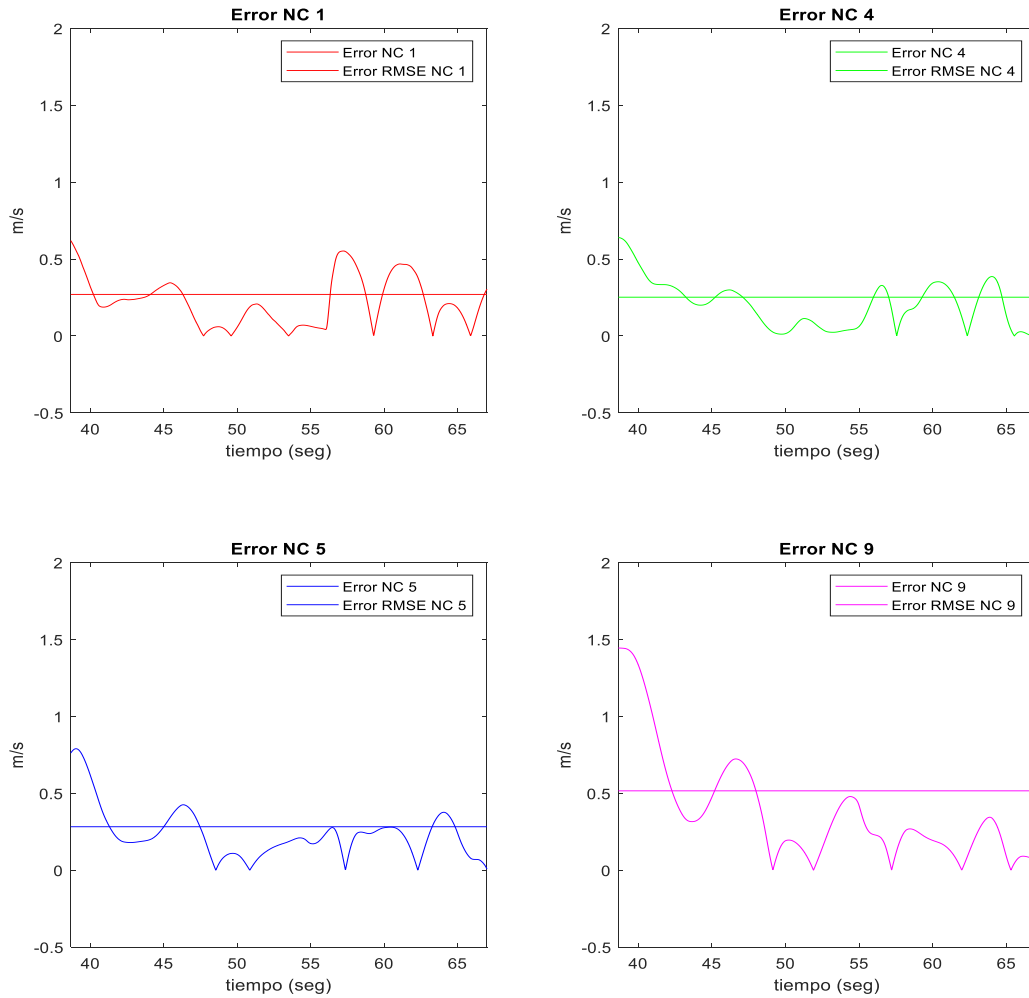


Figura 8.1.4.9. Error absoluto promedio en la velocidad longitudinal al variar el control de horizonte  $N_c = 1, 4, 5, 9$ .

Control de predicción	Error
<b>1</b>	0.2703
<b>4</b>	0.2521
<b>5</b>	0.2826
<b>9</b>	0.5160

Tabla 8.1.4.3. Error RMSE de la velocidad longitudinal al variar el horizonte de predicción.

De lo anterior, resulta que la mejor configuración posible para este escenario es con un valor de control de horizonte de 4; sin embargo, existe una pequeña diferencia en el comportamiento y el error RMSE cuando utilizamos el bloque predeterminado llamado “*Adaptive Cruise Control*” y la herramienta/interfaz de diseño para controladores MPC que nos hace posible modificar el control de horizonte, ambos incluidos en Matlab. Esta diferencia se puede apreciar en la Figura 8.1.4.10.

Resulta evidente que el comportamiento es bastante parecido entre ambos casos, ya que visualmente no presentan una diferencia sustancial. Por lo que mediante el cálculo del error SRME, se determinó que al usar el bloque “ACC” el resultado del comportamiento es ligeramente mejor que cuando se utiliza la interfaz de diseño, obteniendo así un error RMSE de 0.2471 para el primero y de 0.2782 para el segundo durante el mismo intervalo de tiempo  $t = [37,67]$  segundos. Probablemente la causa de esto recae en el uso del bloque ACC, ya que no se tiene control de la velocidad de estimación de los estados de nuestro modelo, mientras que en la herramienta de diseño si se puede manipular dicho valor. En cuanto a similitudes, ambos métodos resultan en los mismos valores óptimos de simulación, es decir, cuando se propone un horizonte de predicción de 40, una robustez media de 0.5 y un control de horizonte de 4 (ya que este es el valor que inicialmente utiliza el bloque ACC), por tanto, para las comparaciones y simulaciones subsecuentes se utilizó el bloque ACC con los valores listados en la Tabla 8.1.4.4.

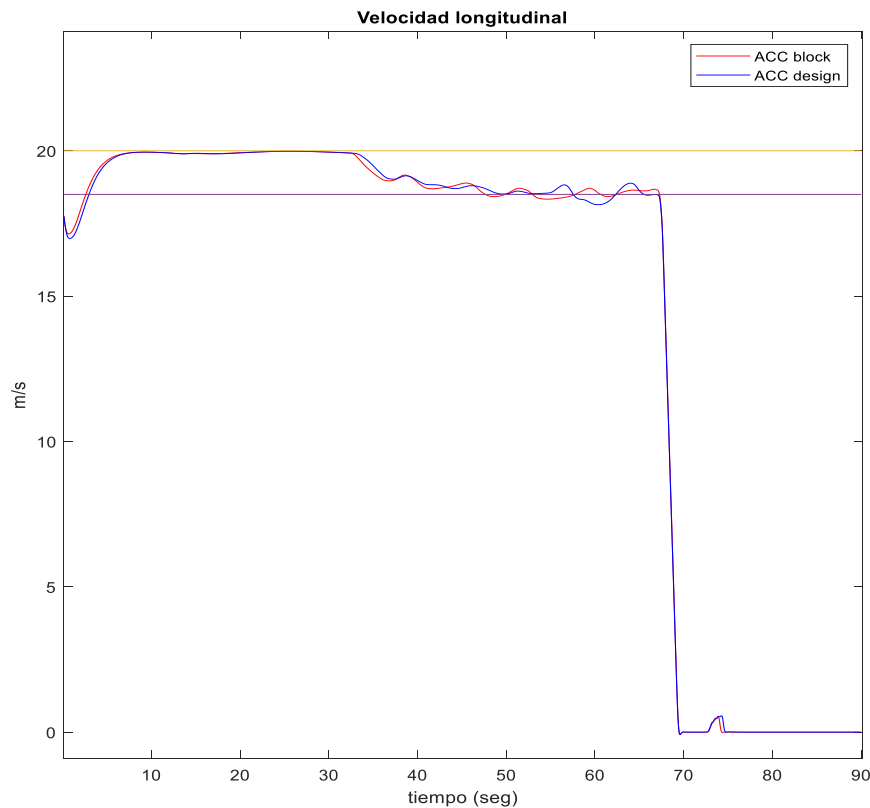


Figura 8.1.4.10. Comparación del desempeño del controlador MPC en dos diferentes acercamientos, bloque e interfaz de diseño.

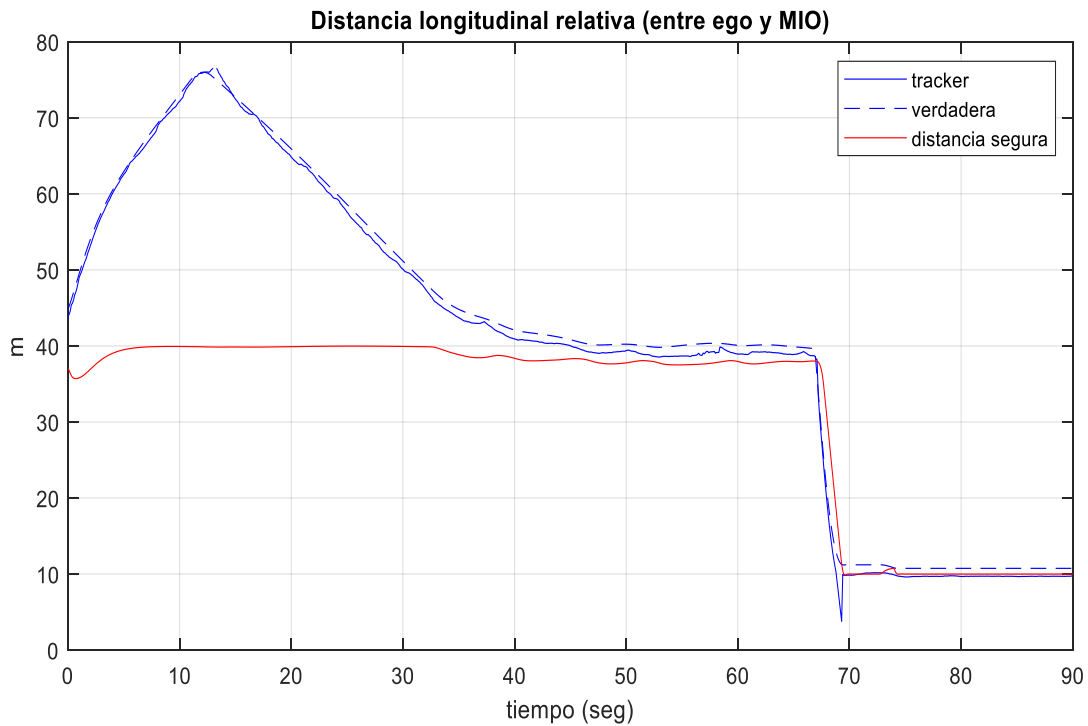


Parámetro	Símbolo	Valor
Tiempo de muestreo	$T_s$	0.1 s
Retraso	$\tau$	0.5 s
Tiempo de avance	H	1 s
Horizonte de predicción	$N_p$	40
Control de horizonte	$N_c$	4
Límite de aceleración superior	$A_{max}$	2 m/s <sup>2</sup>
Límite de aceleración inferior	$A_{min}$	-3 m/s <sup>2</sup>

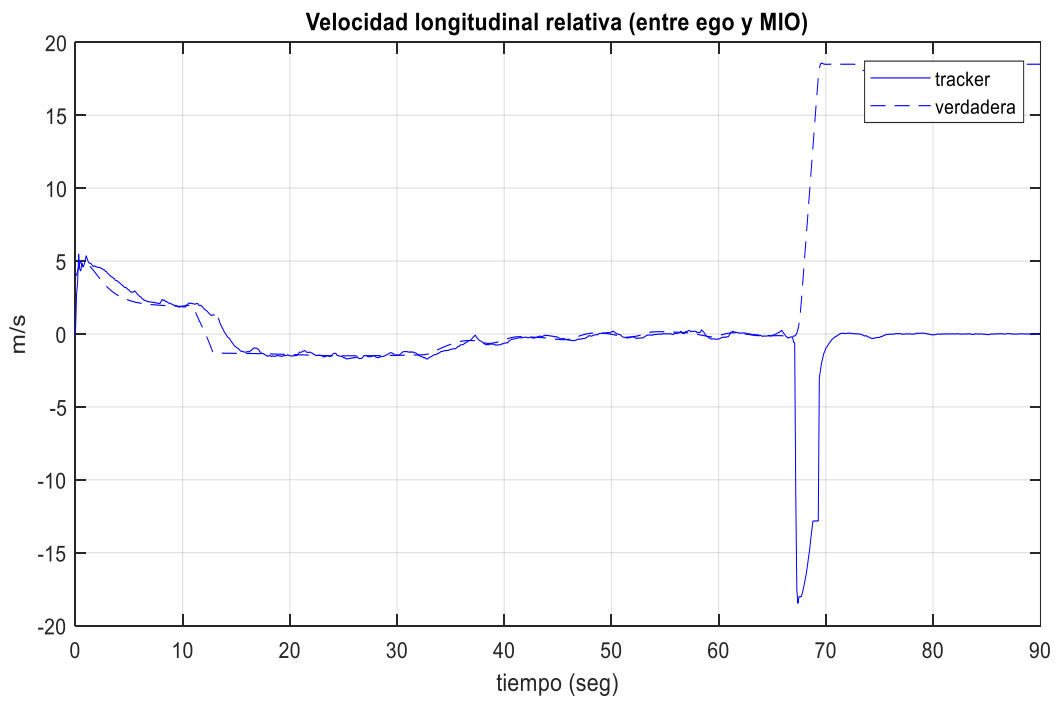
Tabla 8.1.4.4 Parámetros del controlador

A continuación, se presenta en la Figura 8.1.4.11 el desempeño general del controlador MPC de cruceo adaptativo para el escenario 1.

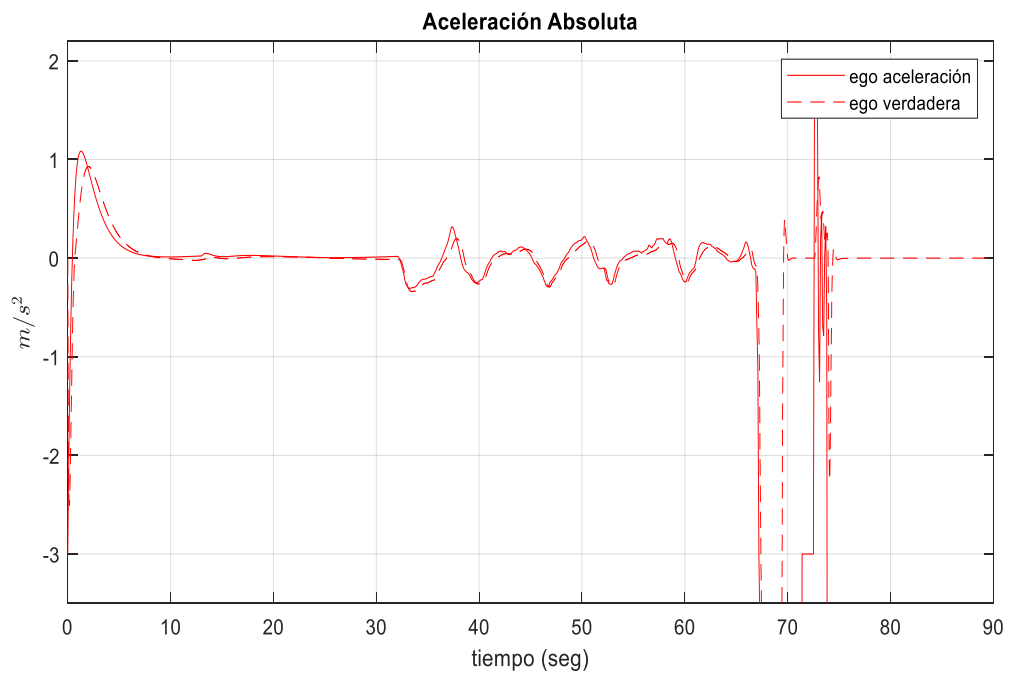
a)



b)



c)



d)

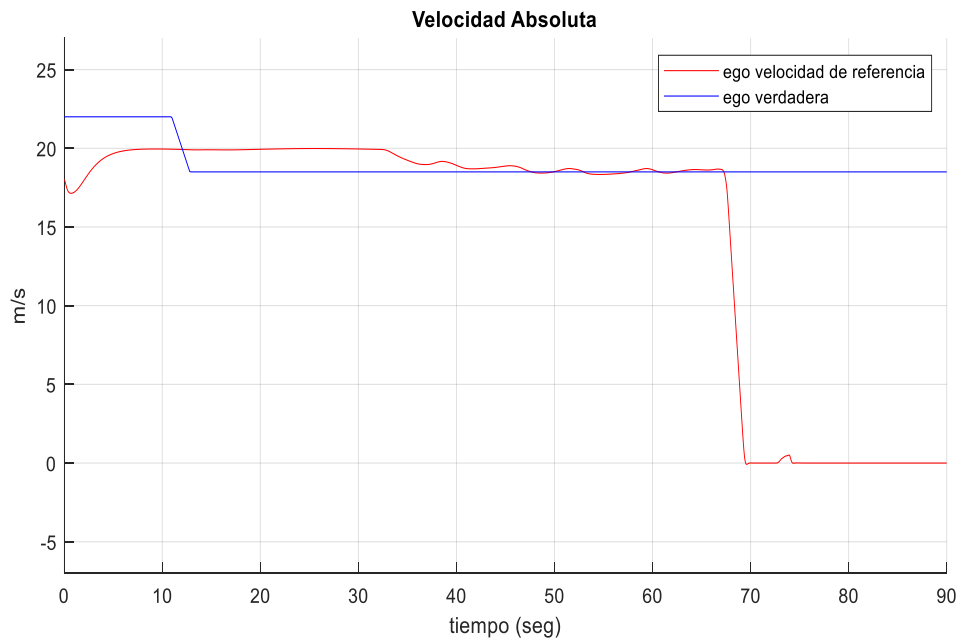


Figura 8.1.4.11. Desempeño del controlador MPC de ACC, a) Distancia longitudinal relativa, b) velocidad longitudinal relativa, c) Aceleración absoluta, d) Velocidad absoluta

De la Figura 8.1.4.11, podemos obtener la siguiente información:

- a) Distancia longitudinal relativa:** en esta gráfica se pueden observar tres diferentes valores de distancia, tal y como se enlistan a continuación: *tracker* (la distancia relativa calculada a partir del algoritmo de procesamiento de visión, basado en las posiciones de los elementos) 9.68 [m], verdadera (distancia de referencia) 10.75 [m], distancia segura 10 [m] (valor extraído al finalizar la simulación que corresponde también al valor de seguridad mínimo). Se observa que durante los 90 [s] de simulación la distancia verdadera se mantiene por encima de la distancia segura. Por otro lado, la distancia calculada al momento de generarse el frenado imprevisto termina por quedar ligeramente por debajo de la distancia segura; sin embargo, el controlador responde de una manera adecuada ante el frenado de emergencia, obteniendo solo 32 [cm] por debajo de la distancia segura y dejando un margen de 9.68 [m] de distancia entre ambos vehículos.
- b) Velocidad longitudinal relativa:** en esta gráfica se muestra la diferencia entre la velocidad longitudinal del vehículo EGO y el vehículo MIO, se puede interpretar que al comienzo de la simulación hay una mayor diferencia entre ambos vehículos; sin embargo, al comenzar a aproximarse esta diferencia se vuelve menor hasta el punto en el que es casi cero, que justamente es uno de los objetivos a cumplir del desempeño del controlador. También, se puede observar que al momento del frenado de emergencia ( $t=67-68$  [s]) hay un comportamiento que difiere entre la velocidad calculada y la verdadera,

esto se debe a que la velocidad calculada tiene como referencia el punto de vista del vehículo EGO, mientras que la verdadera toma como referencia las posiciones reales de ambos vehículos desde la perspectiva del vehículo MIO.

- **c) Aceleración absoluta:** en esta gráfica se visualiza como disminuye la aceleración del vehículo ego en cuanto se acerca al vehículo MIO y continua con una velocidad constante hasta realizar un frenado de emergencia, en donde la aceleración disminuye y comienza a oscilar entre valores muy pequeños para seguir convergiendo a una aceleración casi constante, entre una aceleración positiva y negativa hasta un instante antes del frenado, en cuyo caso se vuelve una aceleración negativa para detenerse por completo hasta llegar a cero.
- **d) Velocidad absoluta:** esta gráfica similar a la de aceleración muestra el cambio en la velocidad cuando el vehículo EGO se aproxima al vehículo MIO y comienza a disminuir hasta frenar por completo al final de la simulación.

Las gráficas a, b, c, d son de vital importancia debido a que engloban y resumen el desempeño del controlador diseñado, estos parámetros son los que nos permitieron comparar los distintos controladores diseñados en este escrito; así como, los diferentes escenarios propuestos en nuestra metodología que se corrieron para cada uno de ellos.

## 8.2 Análisis de Resultados

Una vez determinados los parámetros de simulación a utilizar se procedió a realizar el un análisis sobre su desempeño en cada uno de los escenarios propuestos.

Para poder organizar de una manera adecuada el proceso de comparación de los controladores, para cada uno de ellos se extrajeron puntos de interés y se realizó un índice de ponderación con respecto a nuestros objetivos de control, principalmente centrado en el tiempo de respuesta y oscilación de la señal del controlador con respecto a la referencia deseada. Lo anterior, nos indicaría la velocidad de respuesta del controlador y si cumple con mantener una distancia segura para poder evitar una colisión entre los vehículos y perturbaciones presentes en cada caso, para llevar a cabo la comparación de desempeño se consideraron los siguientes parámetros:

- **V:** sobrepico con respecto a la referencia
- **%V:** porcentaje de sobrepico
- **t1:** tiempo en cruzar por primera vez la referencia
- **$\tau$ :** tiempo en alcanzar el 63% de la señal deseada
- **te:** tiempo en alcanzar una oscilación de la señal estable
- **RMSE:** error de mínimos cuadrados con respecto a la referencia
- **%RMSE:** porcentaje del error RMSE con respecto a la referencia

- a: aceleración [ $\text{m/s}^2$ ]

Conforme se llevaron a cabo las simulaciones de los distintos escenarios, se determinó pertinente incluir un valor adicional a nuestro análisis, este valor se incluyó específicamente como una variante del controlador MPC, es decir, se tiene una versión con un horizonte de predicción de cuarenta y un análisis con un valor de 20.

### 8.2.1 Escenario 1

Como ya se describió anteriormente el escenario 1 presenta un cambio de velocidad y un frenado de emergencia, podemos observar el comportamiento de los cuatro controladores en la Figura 8.2.1 así como algunos valores de interés en la Tabla 8.2.1.1.

En la Tabla 8.2.1.1 se realizó el análisis de algunos datos de interés como la distancia total de frenado, la distancia real entre el vehículo EGO y el vehículo MIO después de haber frenado por completo, la tasa de desaceleración y el rebote que tiene el cálculo de la distancia relativa entre los vehículos, esto para los 4 controladores diseñados.

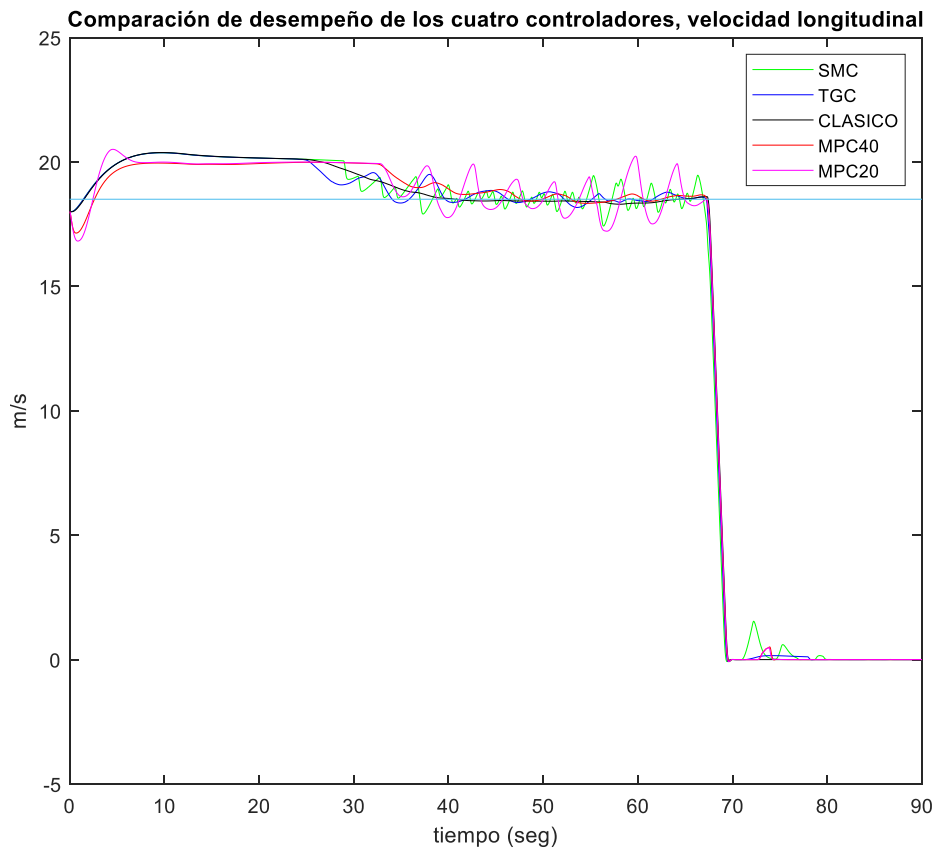


Figura 8.2.1. Escenario 01, comparación de los cuatro controladores y su efecto de la variación de la velocidad longitudinal del vehículo EGO.

ESCENARIO 1										
Vmax	%V	t1	$\tau$	te	RMSE	%RMSE vel	Vmax	%V	a[m/s <sup>2</sup> ]	%Rebote de distancia al frenar
Clásico PI	18.43	0.378	40.6	34.39	42.047	0.151	0.82	0.7	0.32	22.08
SMC	17.91	3.189	36.98	30.64	42.12	0.3306	1.79	1.3	0.59	13.20
CTG	18.34	0.865	34	33.24	50.88	0.2295	1.24	0.3	0.14	23.57
MPC 40	18.44	0.324	47.58	36.14	45.72	0.2521	1.36	2.9	1.32	20.36
MPC 20	17.76	4.000	38.76	34.39	47.24	0.7243	3.92	2.9	1.32	20.62

Tabla 8.2.1.1. Comparación del rendimiento de los controladores diseñados, escenario 01.

ESCENARIO 1									
Controlador	D [m] en t0= 67.1 [s]	D [m] tf= 68.3 [s]	D [m] estado estacionario	V [m/s] en t0	V [m/s] en tf	$\Delta D$ [m]	$\Delta D$ real[m]	$\Delta V$ [m/s]	
Clásico PI	38.83	3.55	9.93	18.6	1.07	35.28	28.9	-17.53	
SMC	38.2	6.13	9.87	17.85	-0.03	32.07	28.33	-17.88	
CTG	37.52	3.55	10.03	18.52	0.28	33.97	27.49	-18.24	
MPC 40	38.74	3.8	9.71	18.63	0.6515	34.94	29.03	-17.98	
MPC 20	38.23	3.83	9.71	18.47	0.405	34.4	28.52	-18.07	

Tabla 8.2.1.2. Comparación del rendimiento de los controladores diseñados, escenario 01.

De la Tabla 8.2.1.2. se puede obtener la siguiente información, los cuatro controladores cumplen con la función específica de regular la velocidad y aceleración tanto para llegar a la velocidad de referencia establecida, como para mantener una distancia segura entre los vehículos. Continuando con el análisis, se obtuvo la velocidad en dos tiempos específicos del frenado, los cuales se denominaron como  $t_0$  y  $t_f$ , siendo que el  $t_0$  es el momento exacto en el que comienza a frenar el vehículo y  $t_f$  es el momento final; sin embargo, en  $t_f$  existe un rebote en la señal con un periodo de tiempo corto, es por eso que la distancia se obtiene de una diferencia ( $\Delta d$ ). Este valor hace referencia a la diferencia en la distancia de  $t_0$  a  $t_f$  incluyendo el rebote de la señal, mientras que  $\Delta d_{real}$  es la diferencia de la distancia sin el rebote de la señal. Se puede observar que los valores de  $\Delta d$  difieren para cada controlador, esto puede deberse tanto a la velocidad a la que viaja el vehículo EGO, como a la tasa de desaceleración de cada uno, es decir, si tomamos el desempeño del controlador MPC, PID y CTG, podemos observar que tienen una velocidad de frenado similar y por tanto un porcentaje de rebote y tasa de desaceleración

parecidas. Es aquí donde el valor de  $\Delta d_{\text{real}}$  cobra relevancia, ya que podemos calcular el tiempo real de frenado y como se observa los cuatro controladores tienen variaciones de centésimas entre cada uno.

ESCENARIO 1									
Controlador	Vmax	%V	t1	$\tau$	te	RMSE	%RMS E vel	Puntuación total	Orden de desempeño
<b>Ponderación (0-1)</b>	1	1	3	3	5	5	5		
<b>Clásico PID</b>	18.43	0.38	121.80	103.17	210.24	0.76	0.76	436.72	<b>2</b>
<b>SMC</b>	17.91	3.19	110.94	91.92	210.60	1.65	1.65	416.77	<b>1</b>
<b>CTG</b>	18.34	0.86	102.00	99.72	254.40	1.15	1.15	458.42	<b>3</b>
<b>MPC 20</b>	18.44	0.32	142.74	108.42	228.60	1.26	1.26	482.28	<b>5</b>
<b>MPC 40</b>	17.76	4.00	116.28	103.17	236.20	3.62	19.58	478.85	<b>4</b>

Tabla 8.2.1.3. Índice de ponderación del rendimiento de los controladores diseñados, escenario 01.

Para poder realizar una conclusión más a detalle sobre el comportamiento de los controladores será necesario analizar los diferentes escenarios propuestos; sin embargo, en este caso particular, de la Tabla 8.2.1.3, podemos decir que los cuatro controladores presentan un comportamiento aceptable y muy similar entre sí al tener un porcentaje de oscilación menos al 2% del valor de referencia y un tiempo de frenado menor a 2 [s]. No obstante, resulta que el controlador SMC presenta un mejor desempeño en este escenario específico.

## 8.2.2 Escenario 2

Como ya se describió anteriormente en este escenario se presenta una reducción de velocidad del vehículo EGO, en la Figura 8.2.2.1 podemos observar el comportamiento de los controladores.

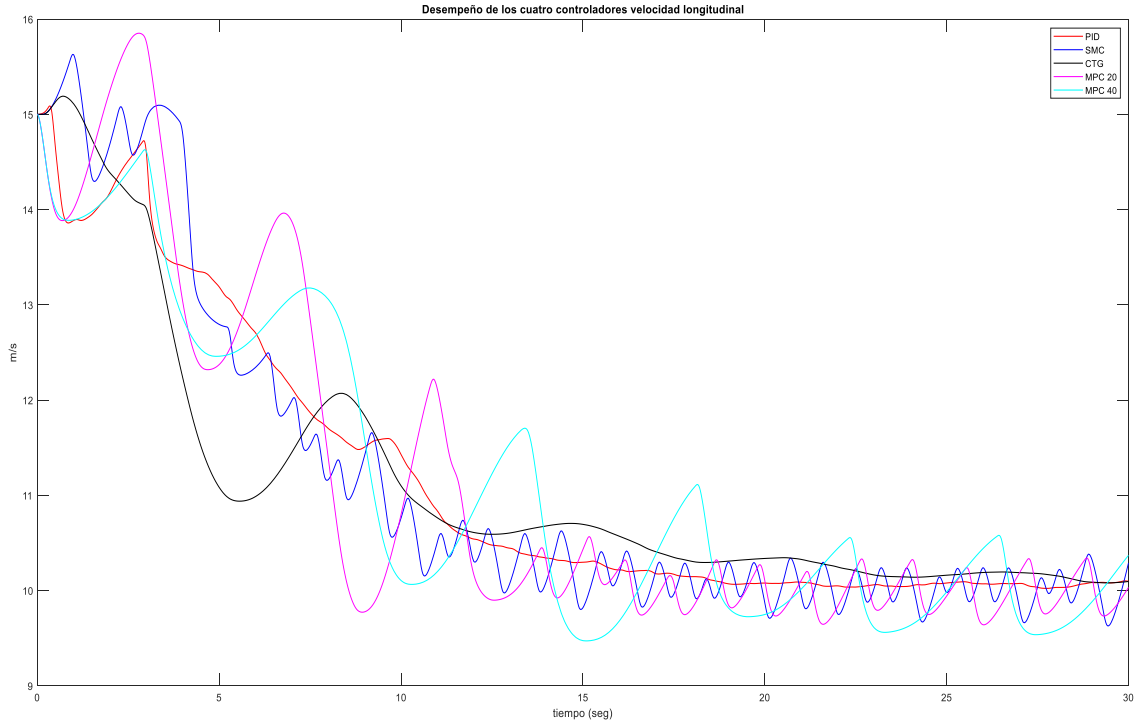
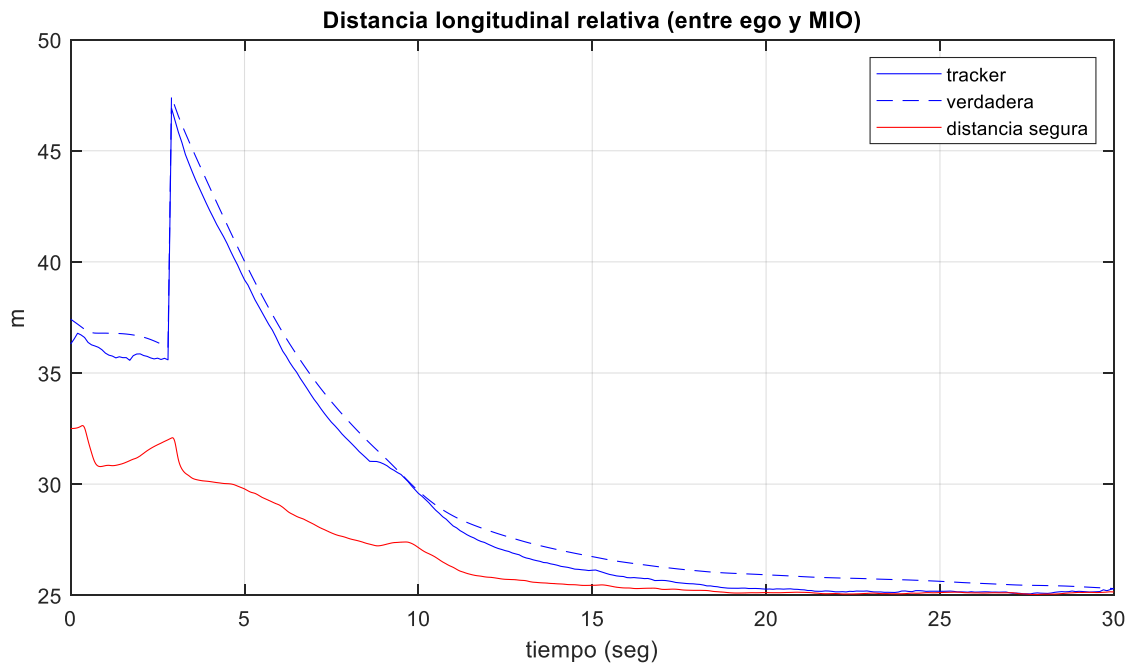


Figura 8.2.2.1. Escenario 02, comparación de los cuatro controladores y su efecto de la variación de la velocidad longitudinal del vehículo EGO.

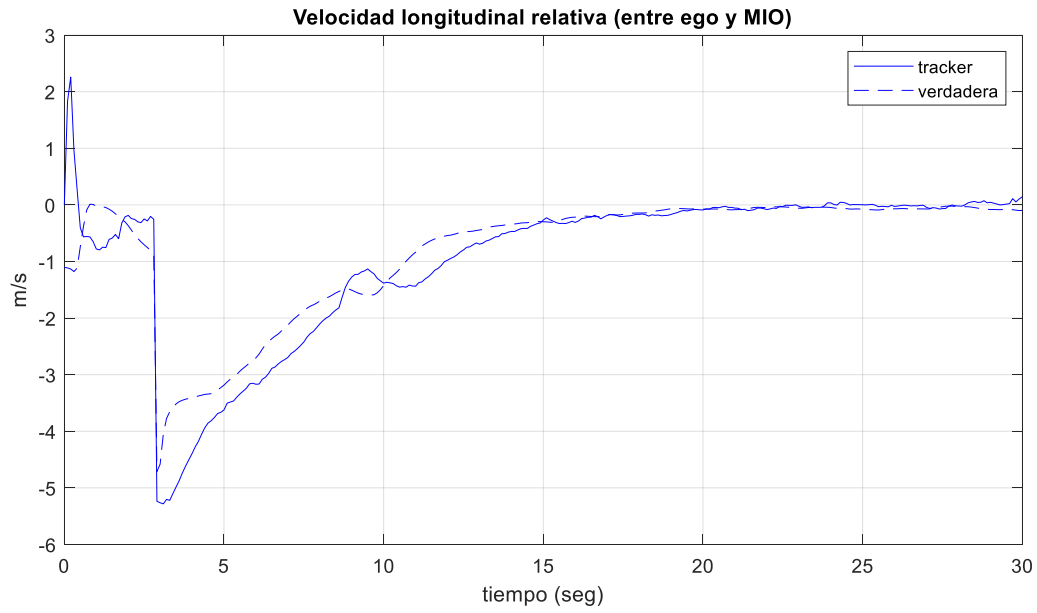
Primeramente, es necesario verificar el desempeño individual de cada controlador ante la reducción de velocidad.

a)

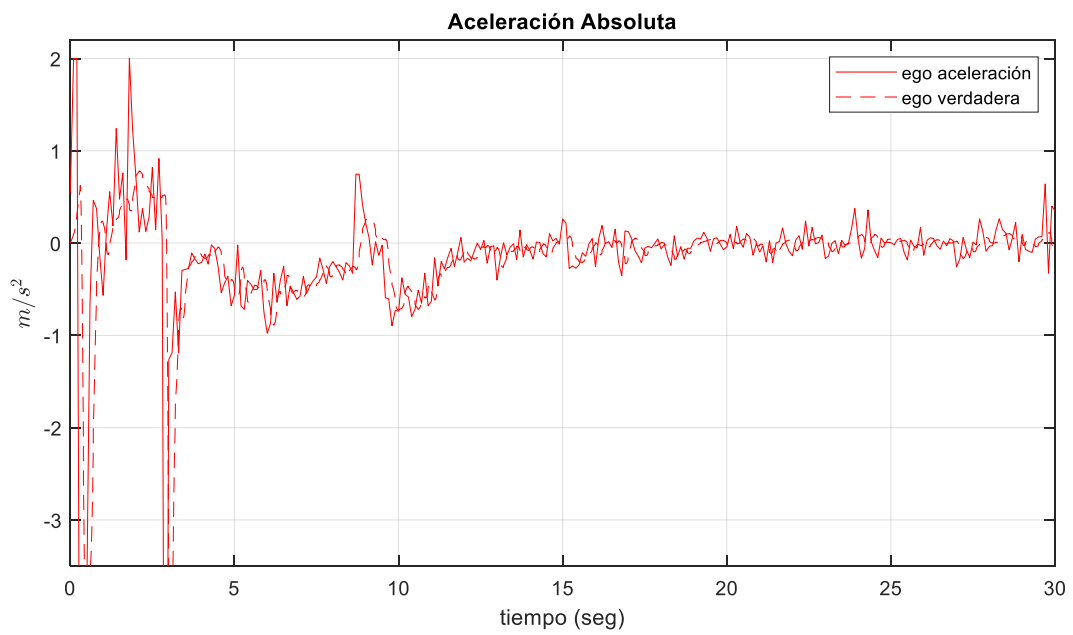




b)



c)



d)

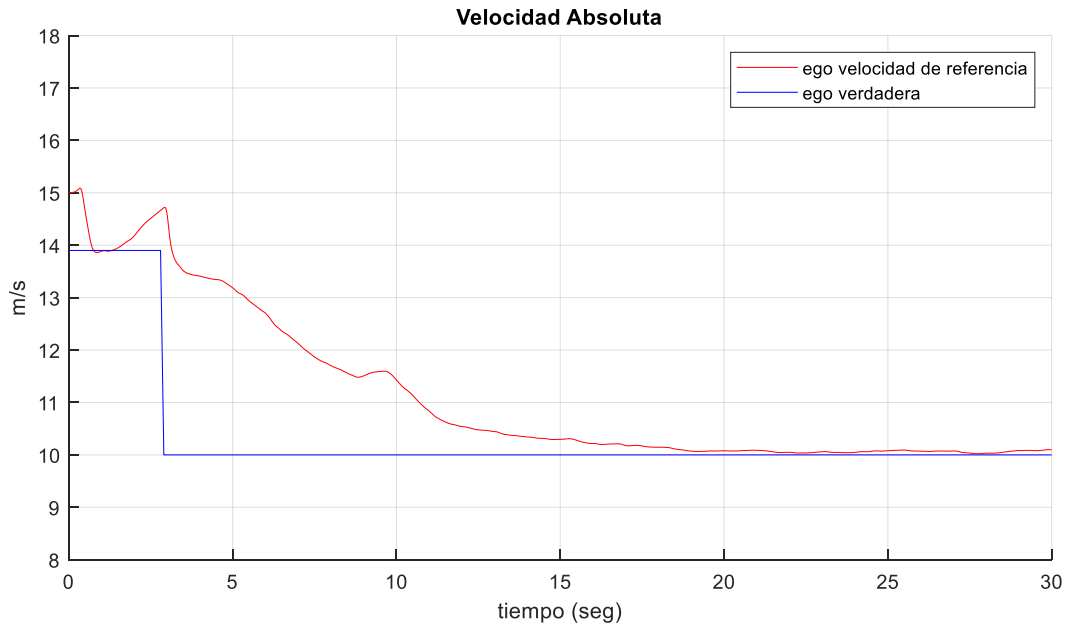
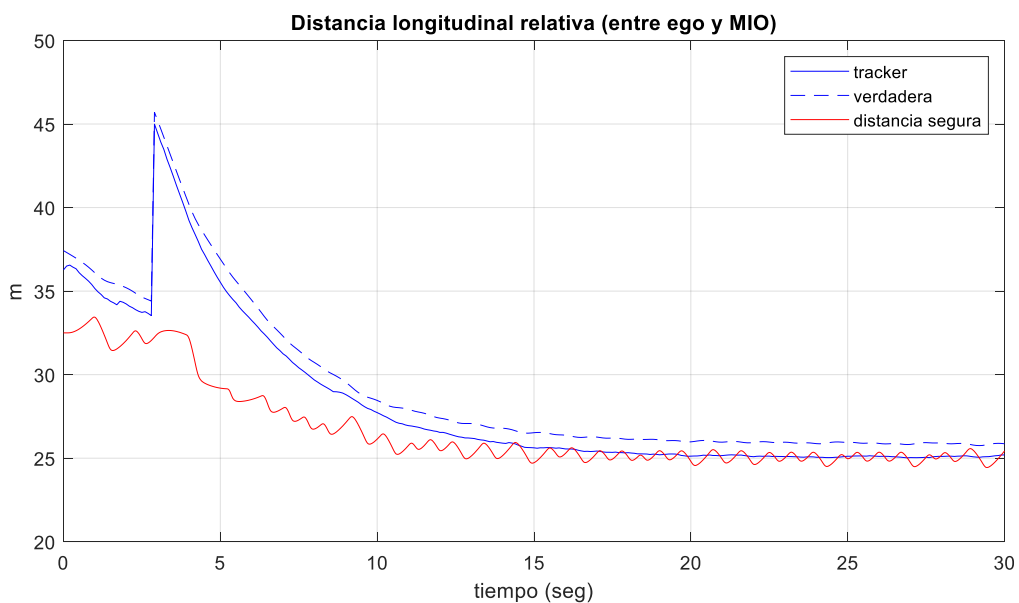


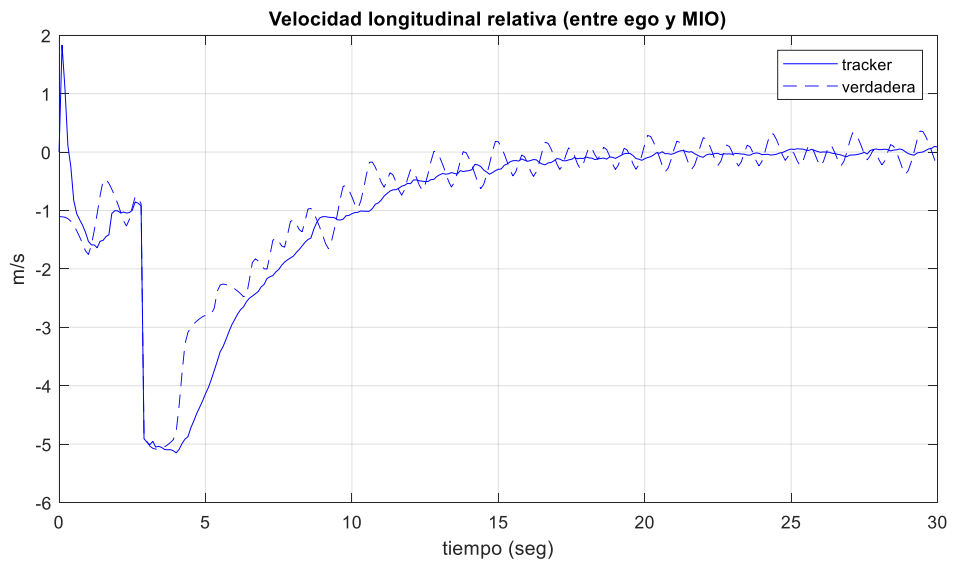
Figura 8.2.2.2. Desempeño del controlador PID de ACC, a) Distancia longitudinal relativa, b) velocidad longitudinal relativa, c) Aceleración absoluta, d) Velocidad absoluta

De la Figura 8.2.2.2 se puede observar que el controlador PID realiza un seguimiento adecuado de la velocidad de referencia, en este caso de 10 [m/s]. No cruza y mantiene una distancia segura mínima durante más de la mitad del recorrido. El controlador evita una colisión con el vehículo MIO y el cambio en la aceleración se da de una manera suave, por lo tanto, satisface nuestro objetivo principal, que es mantener una distancia adecuada entre los vehículos.

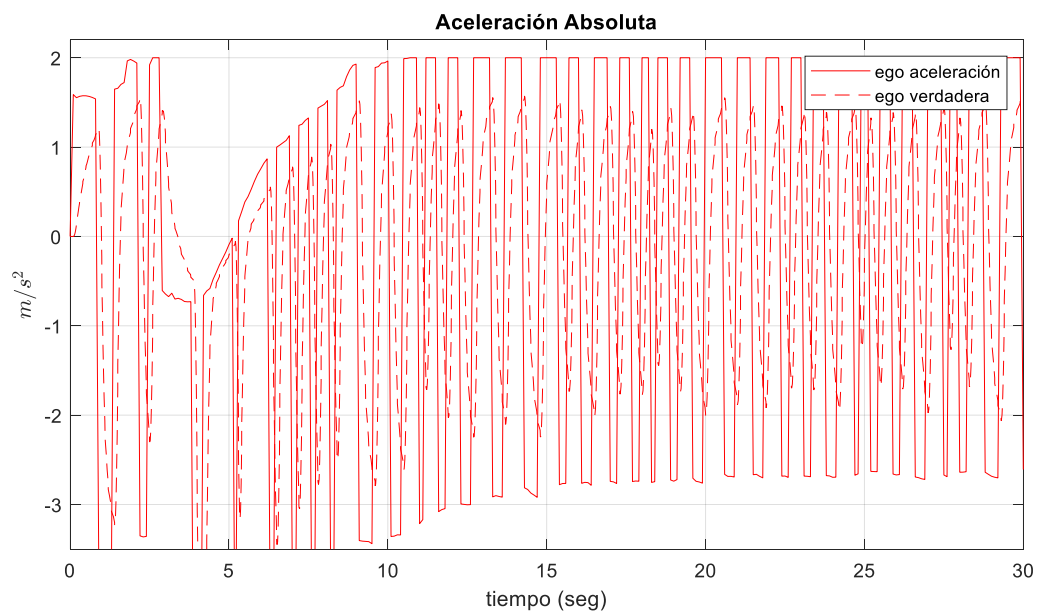
a)



b)



c)



d)

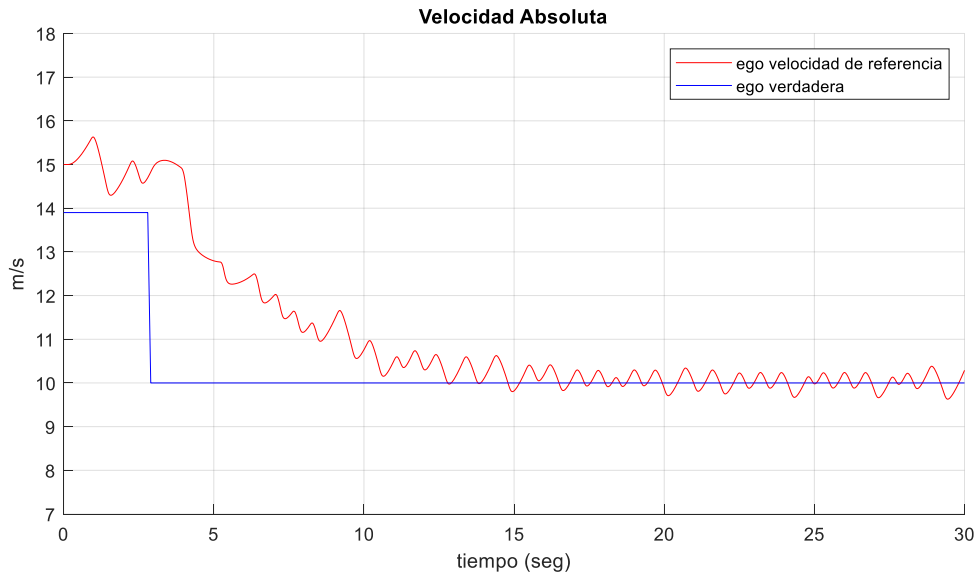
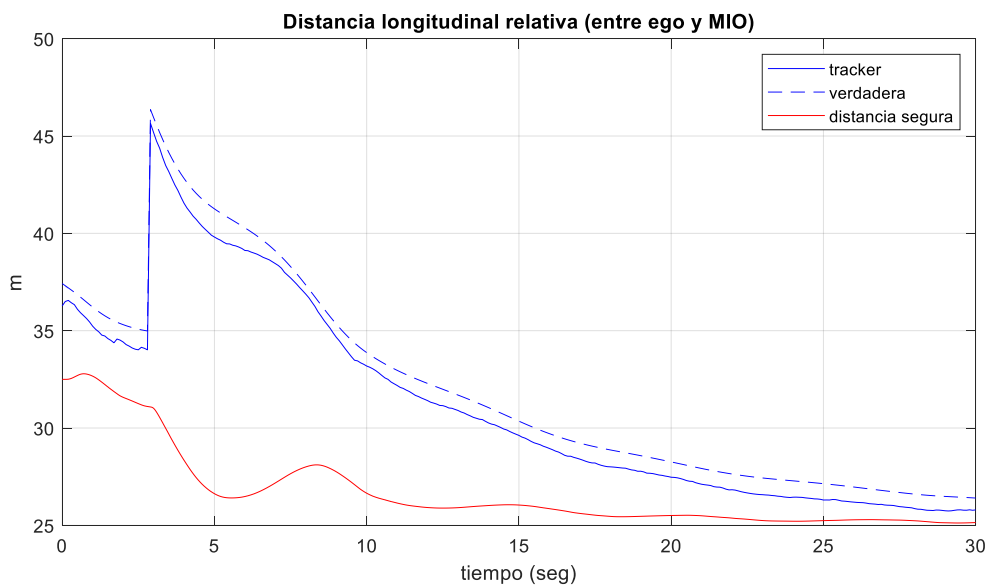


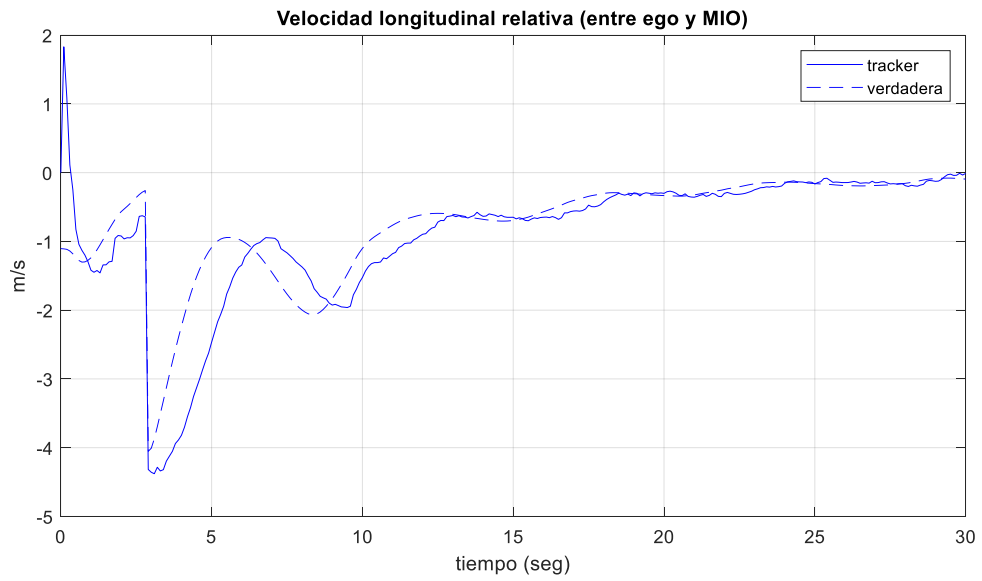
Figura 8.2.2.3. Desempeño del controlador SMC de ACC, a) Distancia longitudinal relativa, b) velocidad longitudinal relativa, c) Aceleración absoluta, d) Velocidad absoluta

De la Figura 8.2.2.3 se puede observar que el controlador SMC realiza un seguimiento adecuado de la velocidad de referencia y prácticamente en todo momento la distancia calculada se mantiene por debajo de la distancia segura, por dicha razón cumple nuestro objetivo de control; sin embargo, sigue manteniendo el comportamiento característico de “chattering” en la señal de la aceleración.

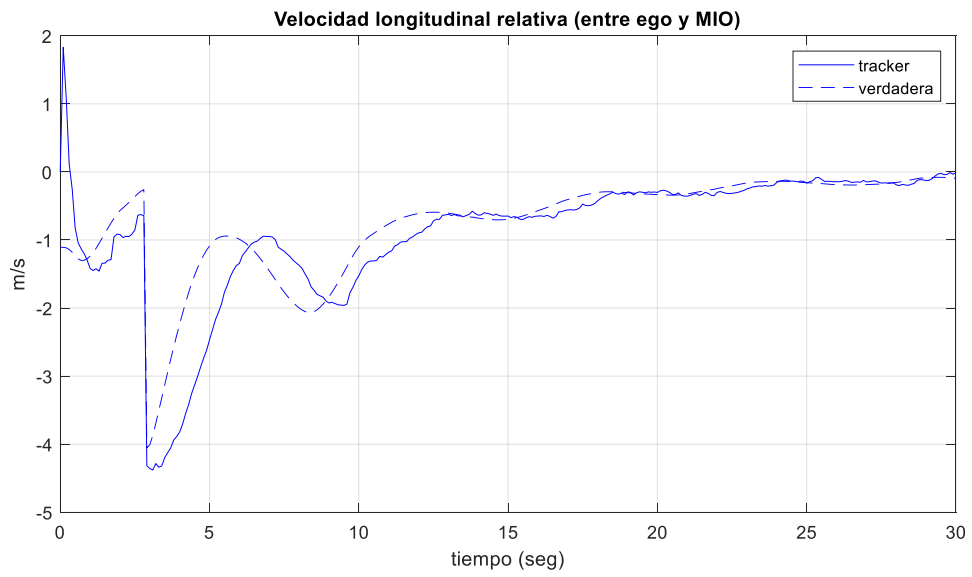
a)



b)



c)



d)

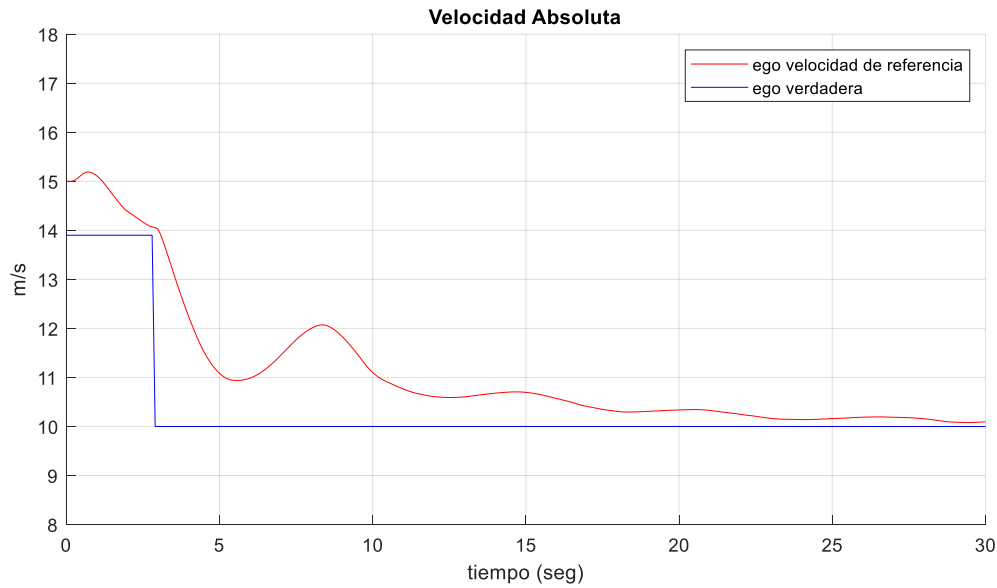
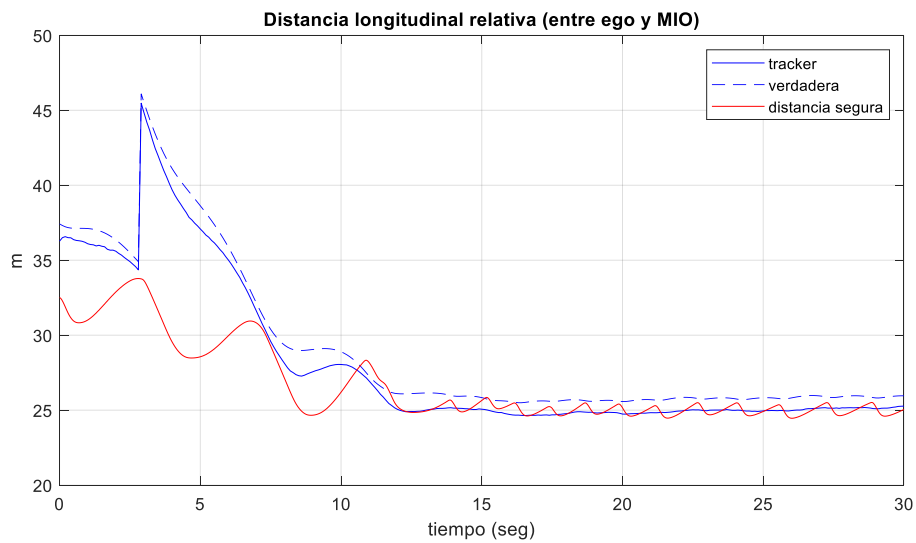


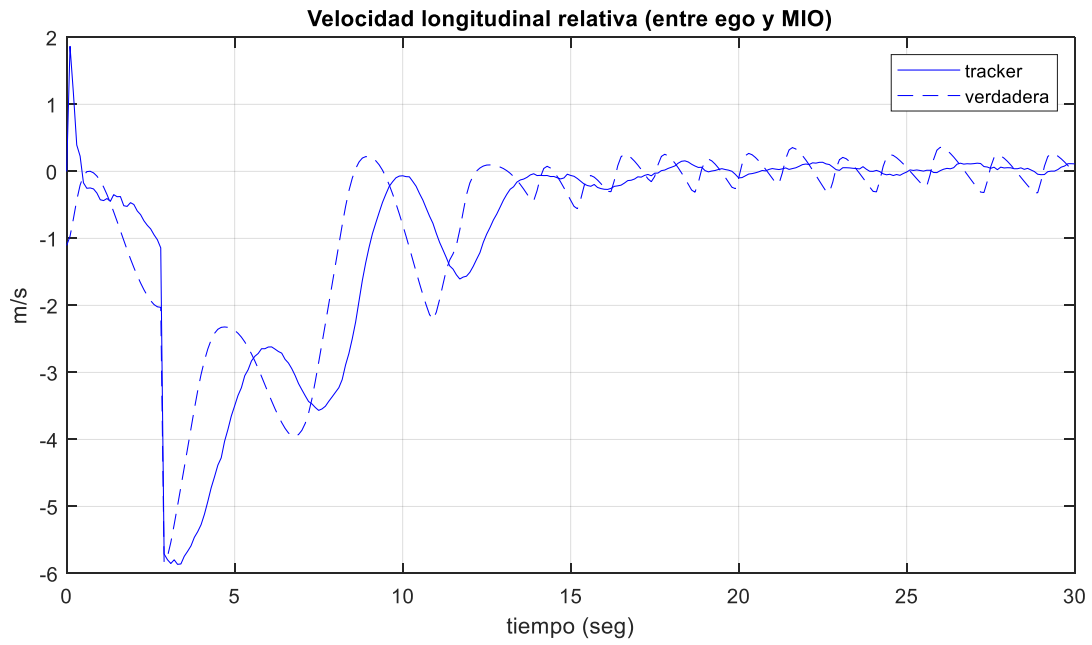
Figura 8.2.2.4. Desempeño del controlador CTG de ACC, a) Distancia longitudinal relativa, b) velocidad longitudinal relativa, c) Aceleración absoluta, d) Velocidad absoluta

De la Figura 8.2.2.4 se puede observar que el controlador CTG tiene un comportamiento bastante bueno, ya que mantiene en todo momento la distancia segura por debajo de la distancia real y calculada, mientras que el comando de la aceleración tiene una tendencia a converger y presenta oscilaciones muy pequeñas; sin embargo, tarda bastante en llegar al valor de referencia.

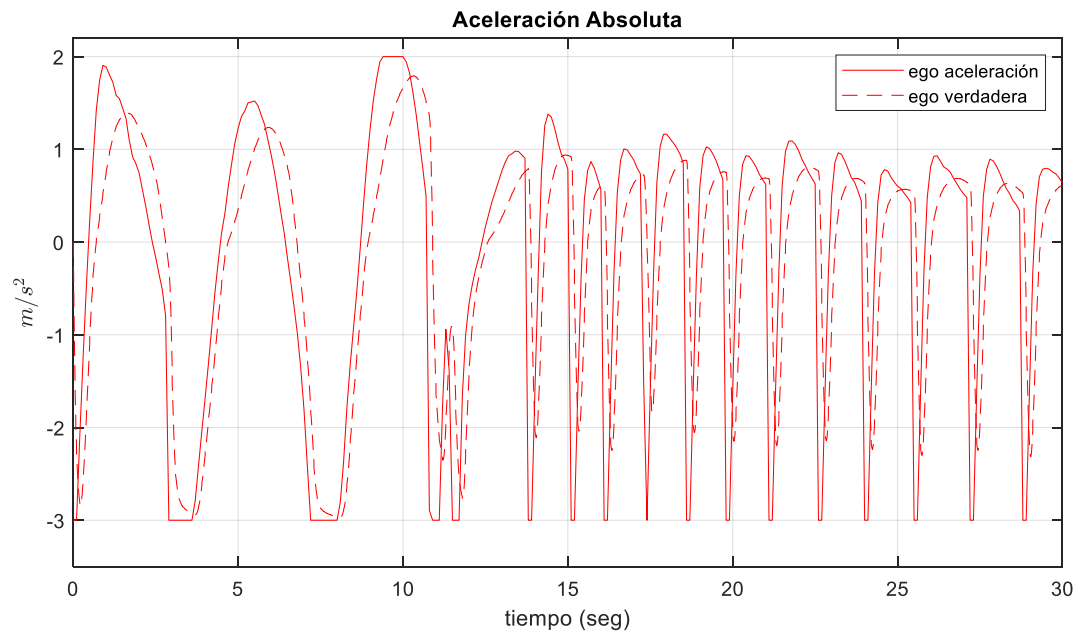
a)



b)



c)



d)

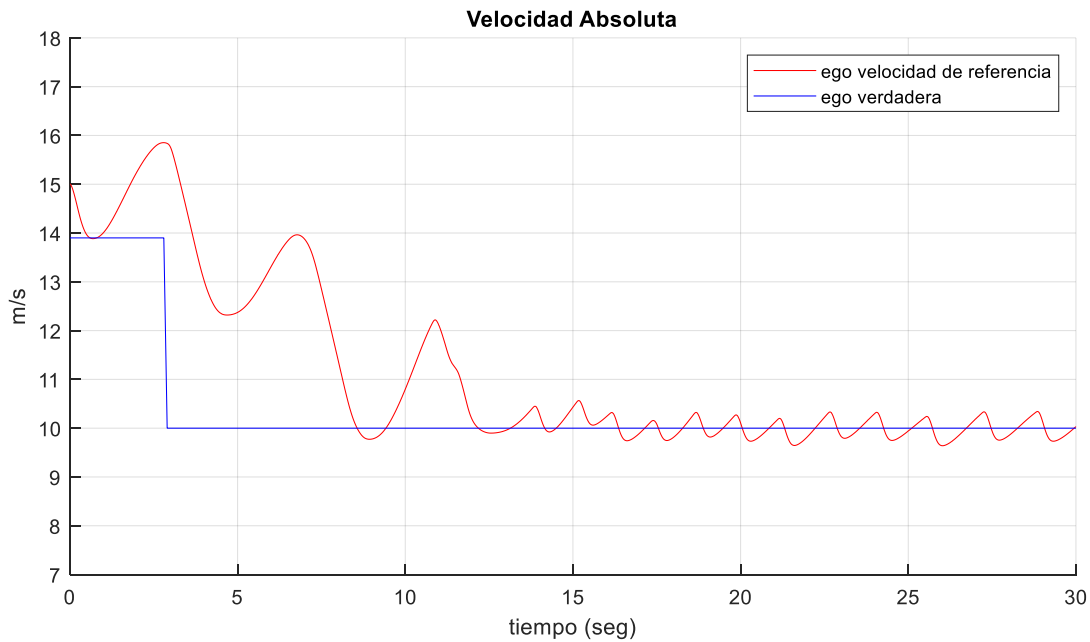


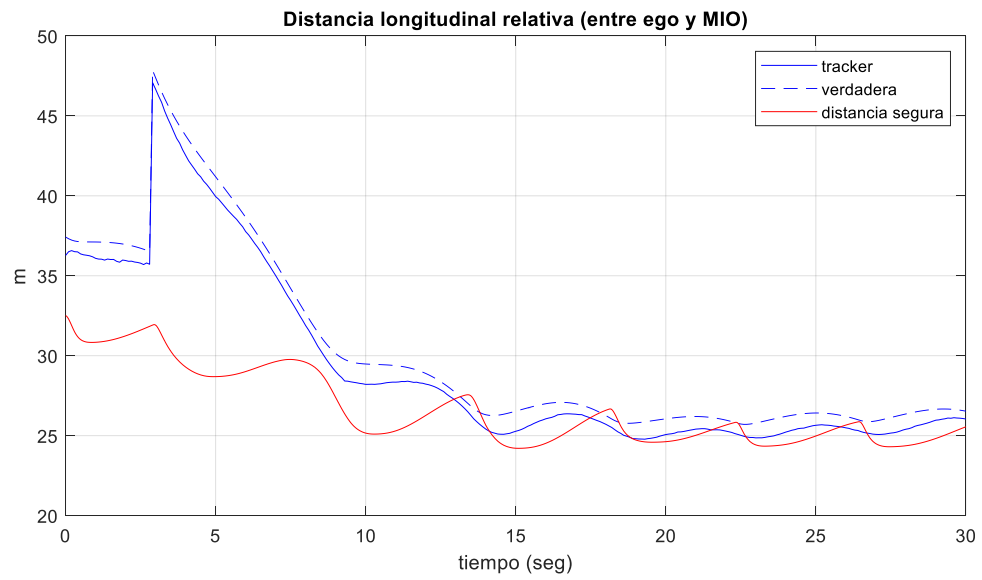
Figura 8.2.2.5. Desempeño del controlador MPC Hp 20 del ACC, a) Distancia longitudinal relativa, b) velocidad longitudinal relativa, c) Aceleración absoluta, d) Velocidad absoluta

De las Figuras 8.2.2.5 y 8.2.2.6 se observa como cambia el comportamiento del controlador MPC dependiendo el horizonte de predicción seleccionado, en este caso al presentarse en el escenario 02 una perturbación en poco tiempo, resulta un mejor comportamiento de la disminución del horizonte de predicción de 40 a 20, ya que de esta forma se obtiene una oscilación de menor amplitud con respecto a la referencia, aunque esto provoca que se produzca un efecto similar al de “chattering” en la señal de aceleración, pero con una amplitud y frecuencia menor que el mostrado en la Figura 8.2.2.3 por el controlador SMC. La diferencia en el comportamiento se puede deber al tiempo en que se presenta la perturbación, es decir que, si la perturbación se presenta en un tiempo corto o muy repentino, un horizonte de predicción pequeño tendrá un mejor comportamiento.

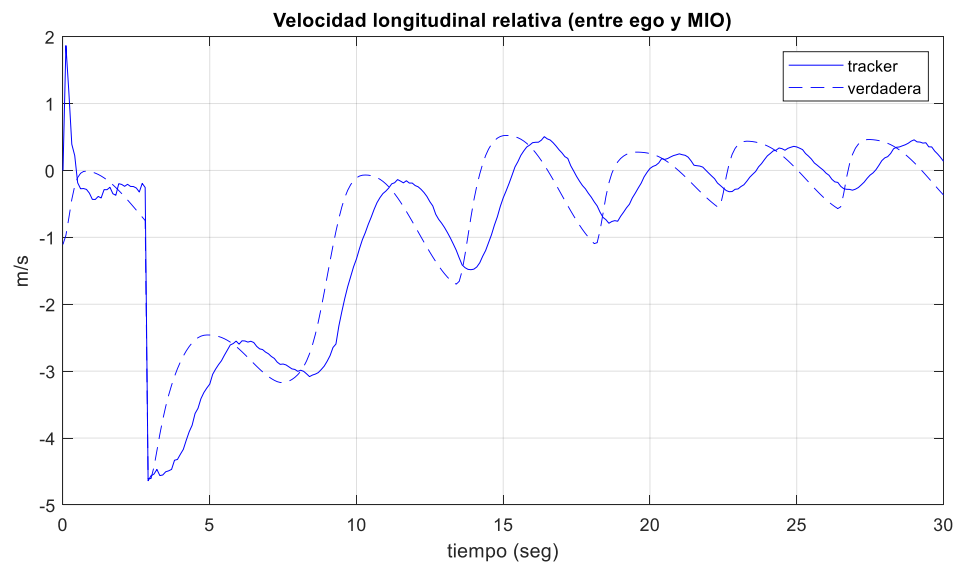
Ahora bien, si tomamos y analizamos puntos de interés de nuestra Figura 8.2.2.1 se obtiene el error RMSE tal y como se muestra en la Figura 8.2.2.7 y la información en la tabla 8.2.2.1.



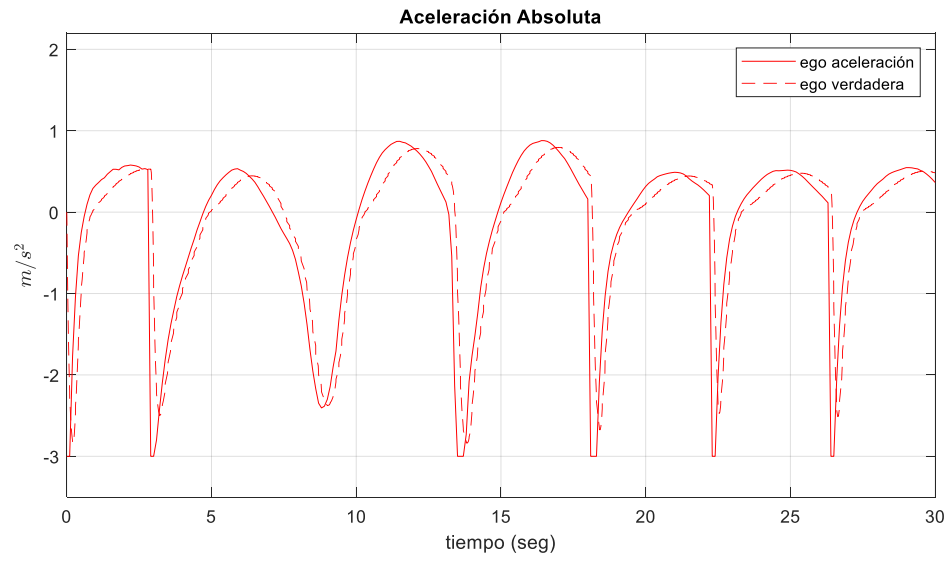
a)



b)



c)



d)

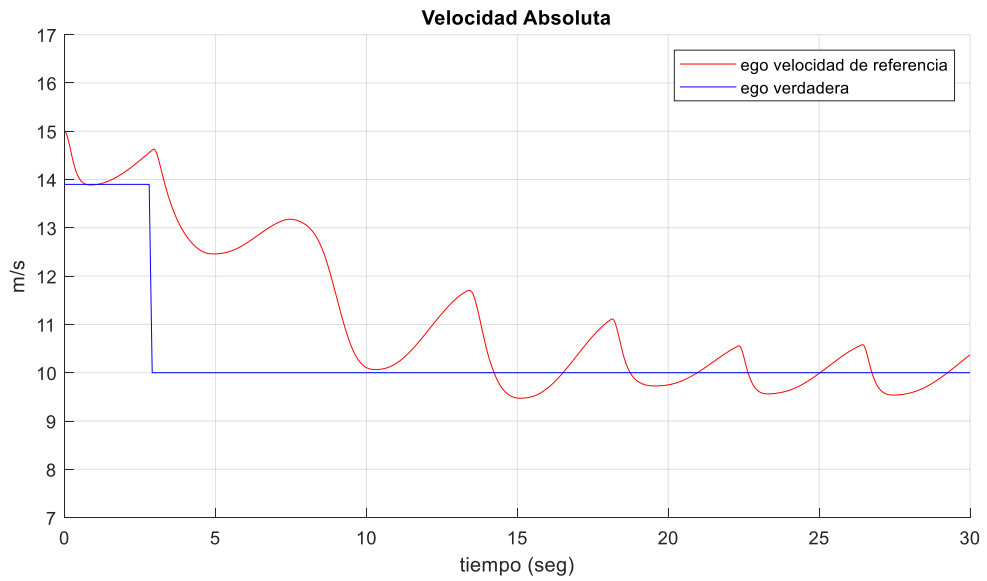


Figura 8.2.2.6. Desempeño del controlador MPC Hp 40 del ACC, a) Distancia longitudinal relativa, b) velocidad longitudinal relativa, c) Aceleración absoluta, d) Velocidad absoluta

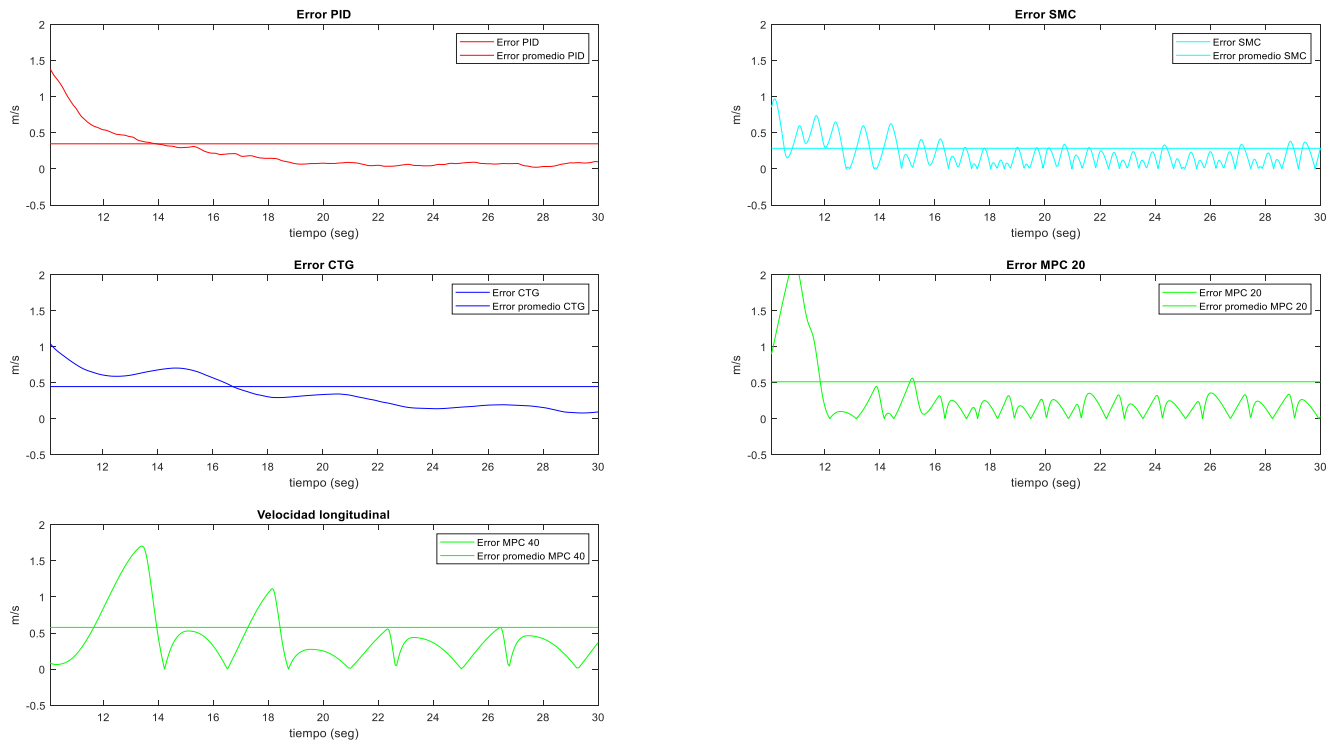


Figura 8.2.2.7. Variación del error RMSE de cada controlador con respecto al valor de referencia.

ESCENARIO 2								
Controlador	Vmax	%V	t1	$\tau$	te	RMSE	%RMSE vel	
Clásico PID	Na	Na	10.071	10	10	0.347	3.47	
SMC	Na	Na	12.84	6.70	15.2	0.2847	2.85	
CTG	Na	Na	>30	4.25	21.61	0.4475	4.48	
MPC 20	9.77	2.3	8.54	7.84	14.52	0.5145	5.15	
MPC 40	9.47	5.3	14.22	8.90	16.5	0.5784	5.78	

Tabla 8.2.2.1. Comparación del rendimiento de los controladores diseñados, escenario 02.

De la Tabla 8.2.2.1 podemos observar que el sobrepico, el tiempo en el que cruza la referencia por primera ocasión y el tiempo que tarda en alcanzar el 63% del valor de referencia, resulta bastante variable en los cuatro controladores, ya que algunos no tienen un comportamiento claro en el que sobrepase el valor de referencia como el SMC, PID y el CTG; por esta razón, los parámetros que tendrán una relevancia mayor son aquellos que faciliten la

comparación. Ahora bien, cabe señalar estos parámetros son aquellos que tienen que ver en la velocidad de adaptación de los controladores, así como el rango de valores que presentan al ajustarse a la referencia o perturbación.

Para lograr lo anterior se implementó un índice ponderado con el propósito de evaluar el desempeño de los controladores, para ello se realizó una ponderación de 0-1 para identificar los valores de mayor relevancia. Tal y como se sigue en la Tabla 8.2.2.2.

EVALUACIÓN DEL CONTROLADOR									
Controlador	Vmax	%V	t1	$\tau$	te	RMSE	%RMSE vel	Puntuación total	Orden de Desempeño
<b>Ponderación (0-5)</b>	1	1	3	3	5	5	5		-
<b>Clásico PID</b>	0.00	0.00	30.21	30.00	50.00	1.73	17.35	129.29	<b>1</b>
<b>SMC</b>	0.00	0.00	38.52	20.10	76.00	1.42	14.24	150.28	<b>3</b>
<b>CTG</b>	0.00	0.00	64.83	12.75	108.05	2.24	22.38	210.24	<b>5</b>
<b>MPC 20</b>	9.77	2.30	25.62	23.52	72.60	2.57	25.73	150.04	<b>2</b>
<b>MPC 40</b>	9.47	5.30	42.66	26.70	82.50	2.89	28.92	183.67	<b>4</b>

Tabla 8.2.2.2. Índice de ponderación para seleccionar el controlador con el mejor desempeño, escenario 02.

De la Tabla 8.2.2.2 se concluye que el mejor desempeño lo tiene el controlador PID, seguido por muy poca diferencia con respecto al MPC. De manera similar, el SMC también presenta un comportamiento muy bueno, exceptuando el tiempo de respuesta como una de sus características más débiles. Como podemos observar existe poca diferencia entre los controladores para este escenario con la excepción del PID y todos ellos cumplen con el objetivo de control.

### 8.2.3 Escenario 3

Como se describió anteriormente, en este escenario se presentan dos cambios de velocidad, lo cual implica que se tendrán dos velocidades de referencia, en la Figura 8.2.3.1 se observa una comparación entre los controladores de como varia la velocidad longitudinal en cada uno de ellos.

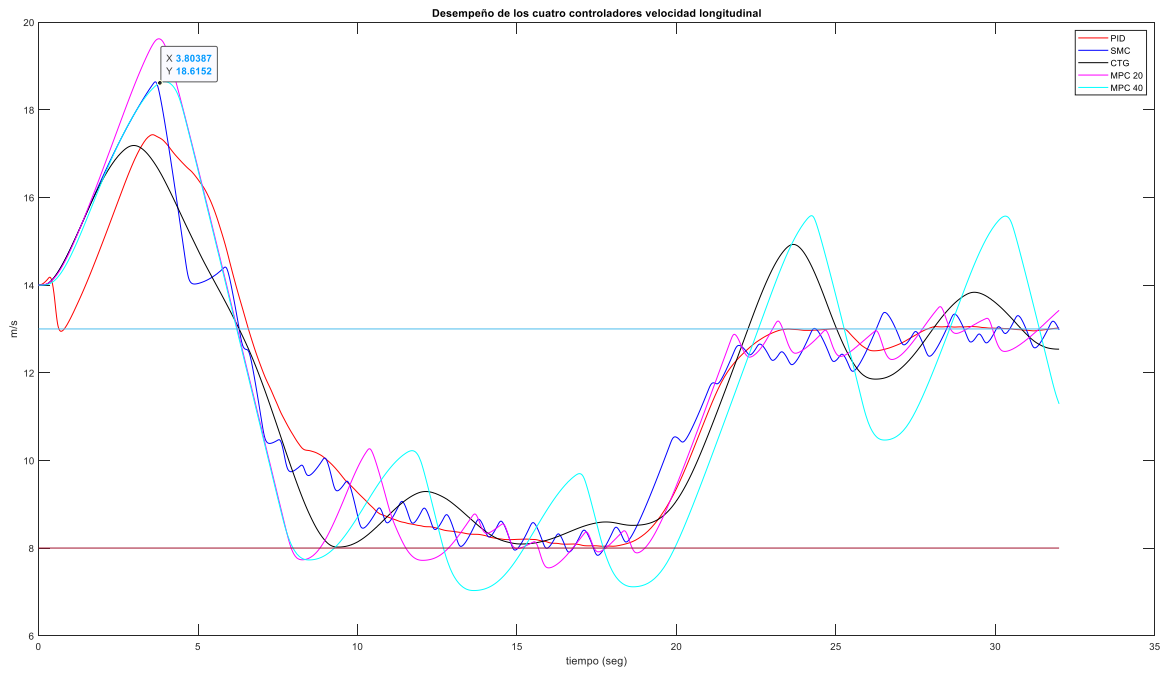
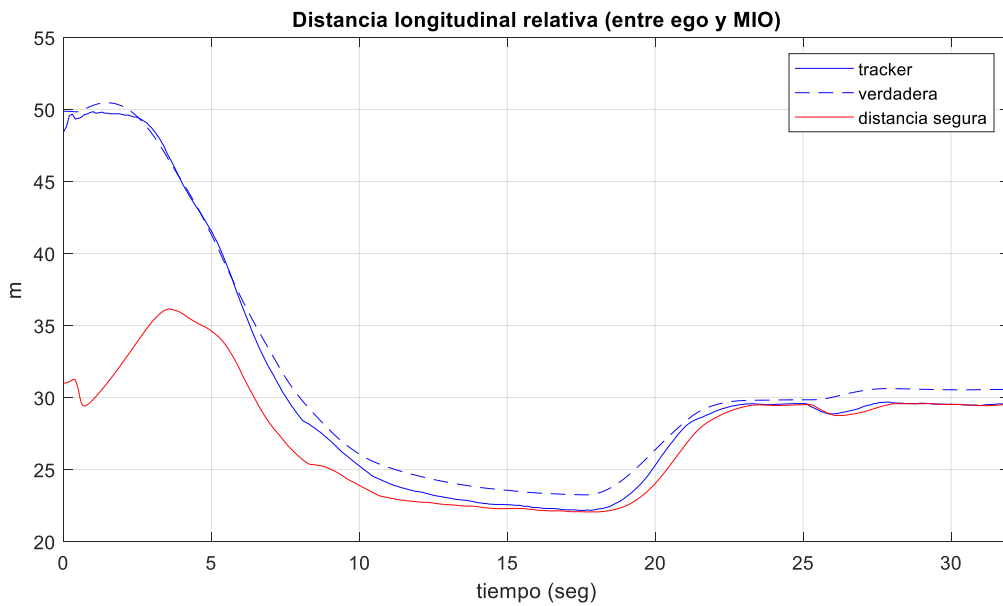


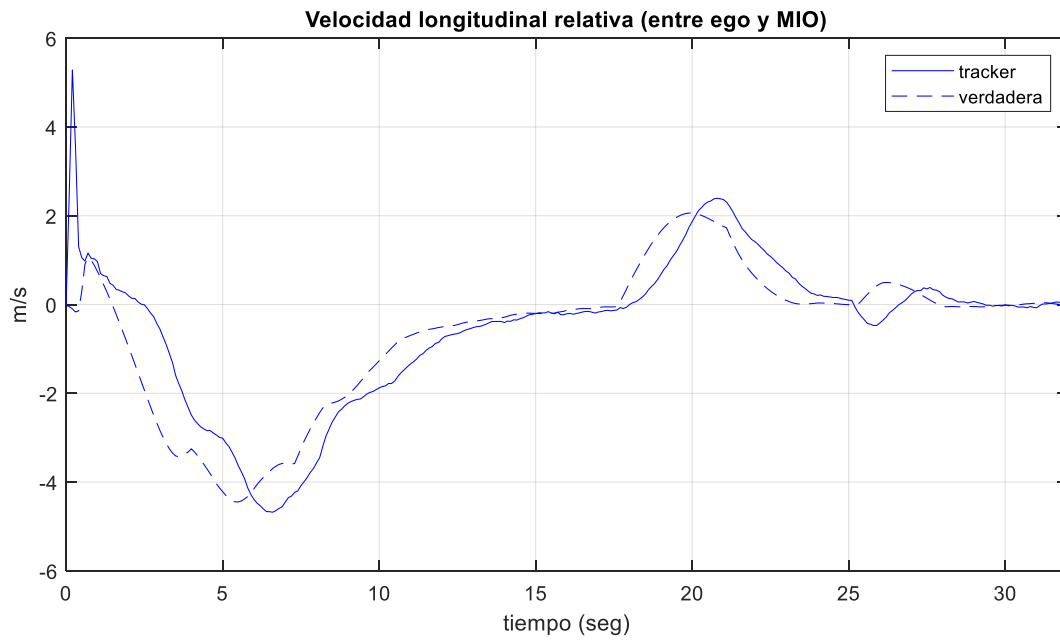
Figura 8.2.3.1 Escenario 03, comparación de los cuatro controladores y su efecto de la variación de la velocidad longitudinal del vehículo EGO.

Si verificamos el comportamiento de cada controlador por separado se obtiene lo siguiente:

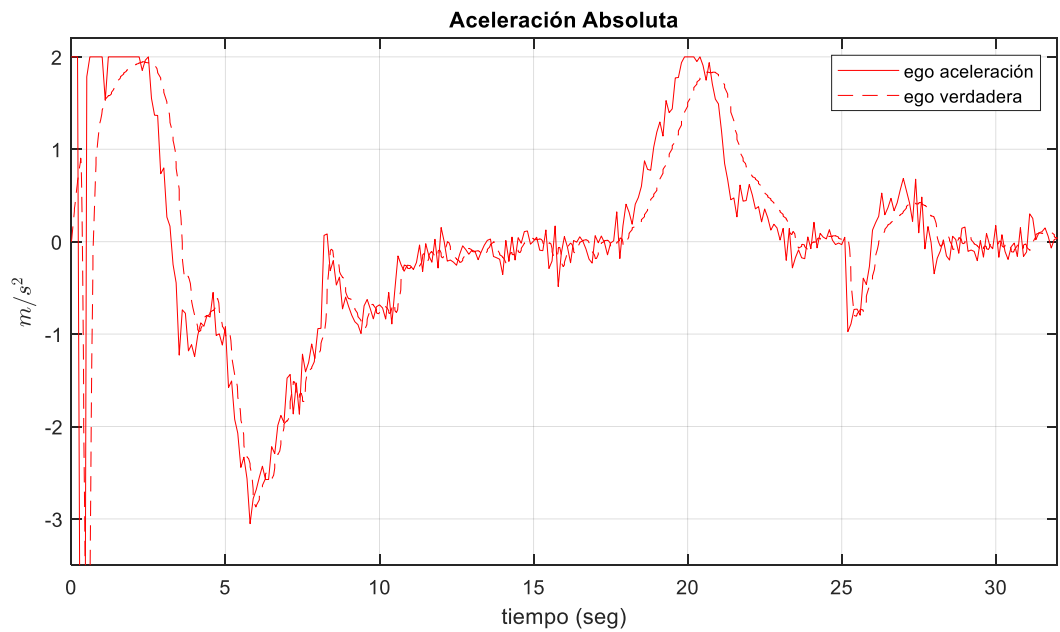
a)



b)



c)



d)

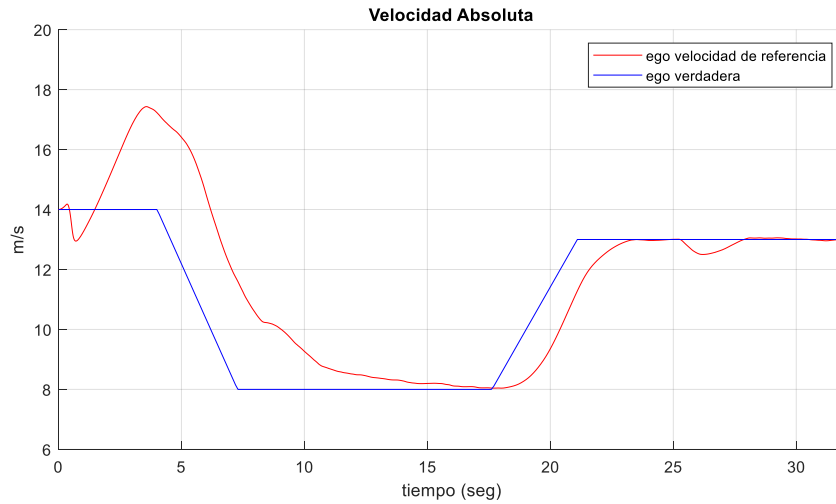
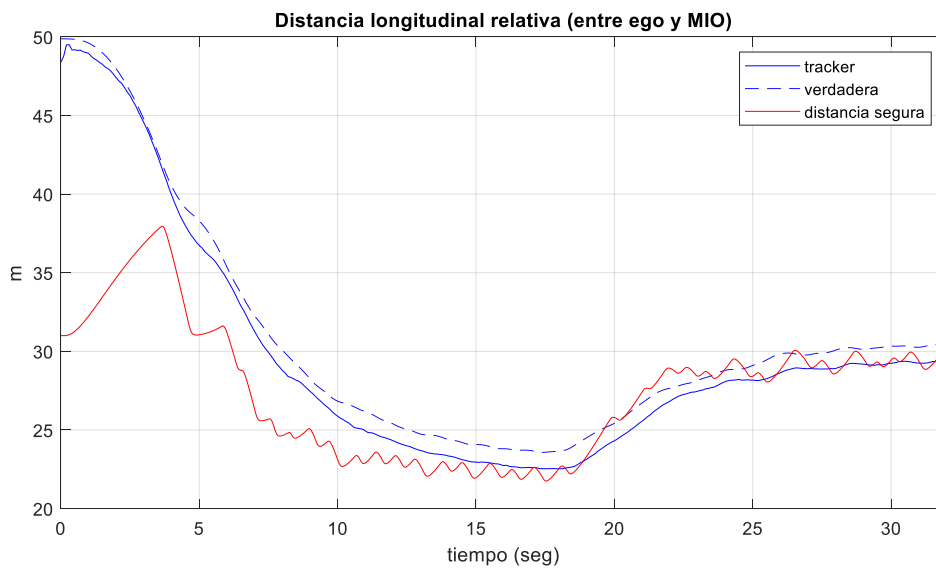


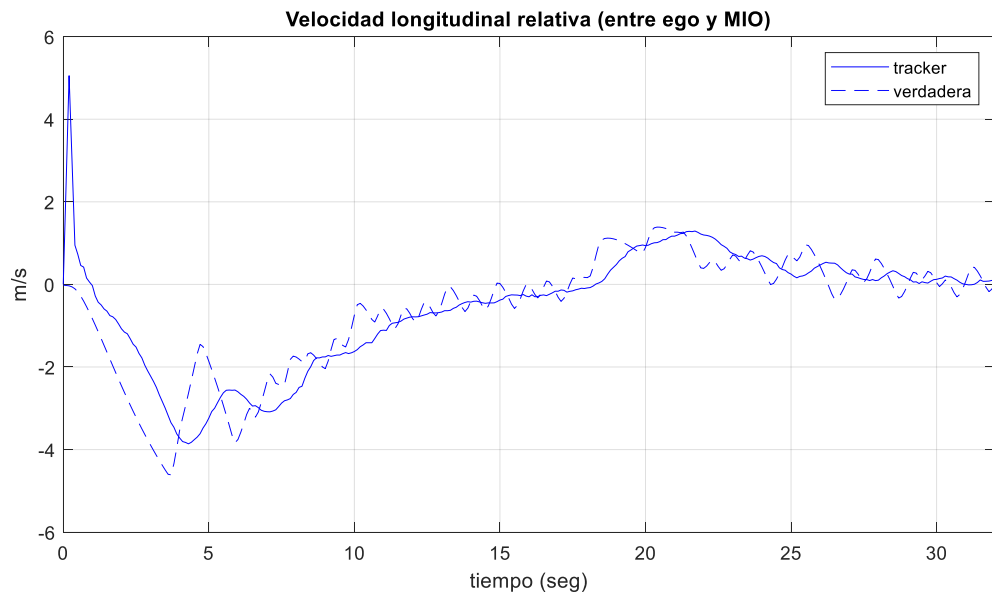
Figura 8.2.3.2. Desempeño del controlador PID de ACC, a) Distancia longitudinal relativa, b) velocidad longitudinal relativa, c) Aceleración absoluta, d) Velocidad absoluta

De la Figura 8.2.3.2 podemos observar que el controlador PID tiene un comportamiento bastante bueno con respecto a los cambios de velocidad entre el vehículo EGO y el MIO, esto debido a que el cambio de velocidad cuando requiere mantener una distancia constante decremента adecuadamente. Posteriormente, vuelve a incrementarse; dicho aumento en la velocidad es adecuado, ya que tiene un valor de sobre pico prácticamente mínimo comparado con el resto de controladores. Además, cumple en todo momento con mantener una distancia mayor a la distancia segura.

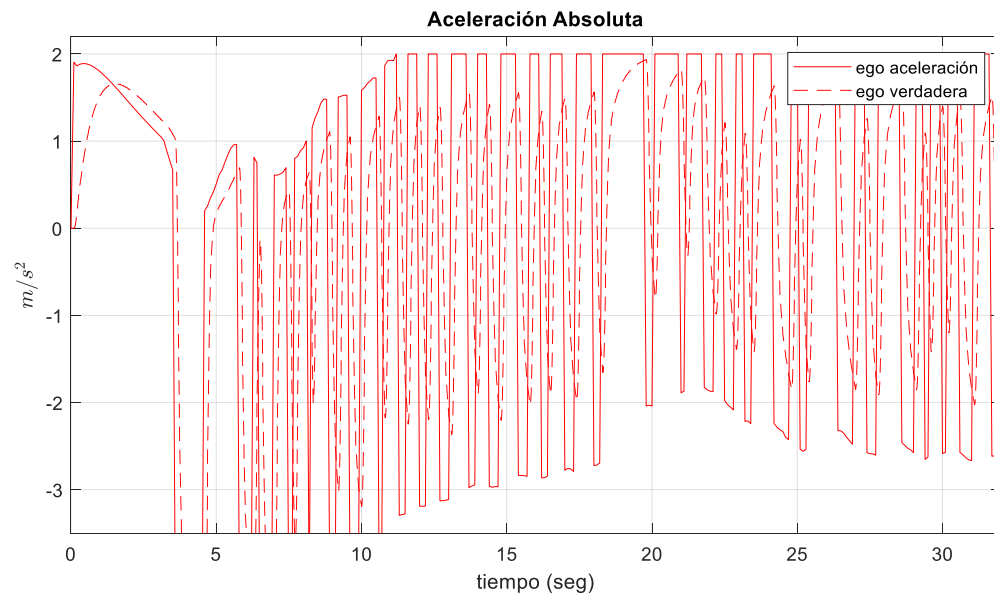
a)



b)



c)





d)

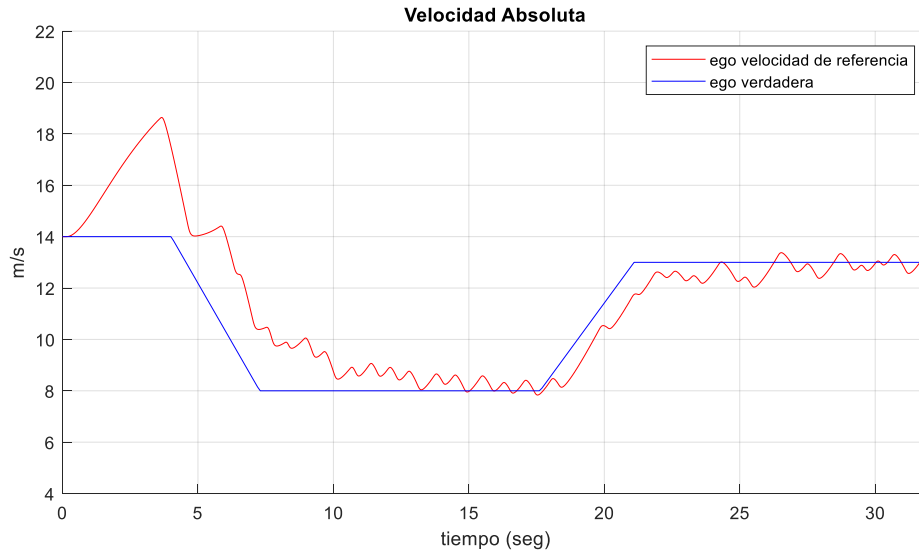
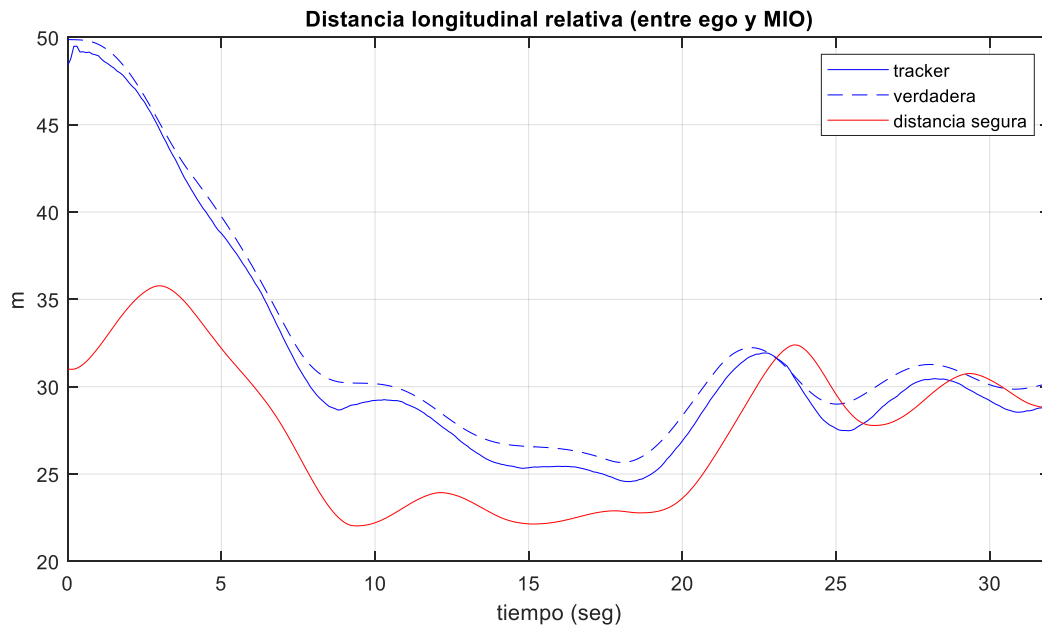


Figura 8.2.3.3. Desempeño del controlador SMC de ACC, a) Distancia longitudinal relativa, b) velocidad longitudinal relativa, c) Aceleración absoluta, d) Velocidad absoluta

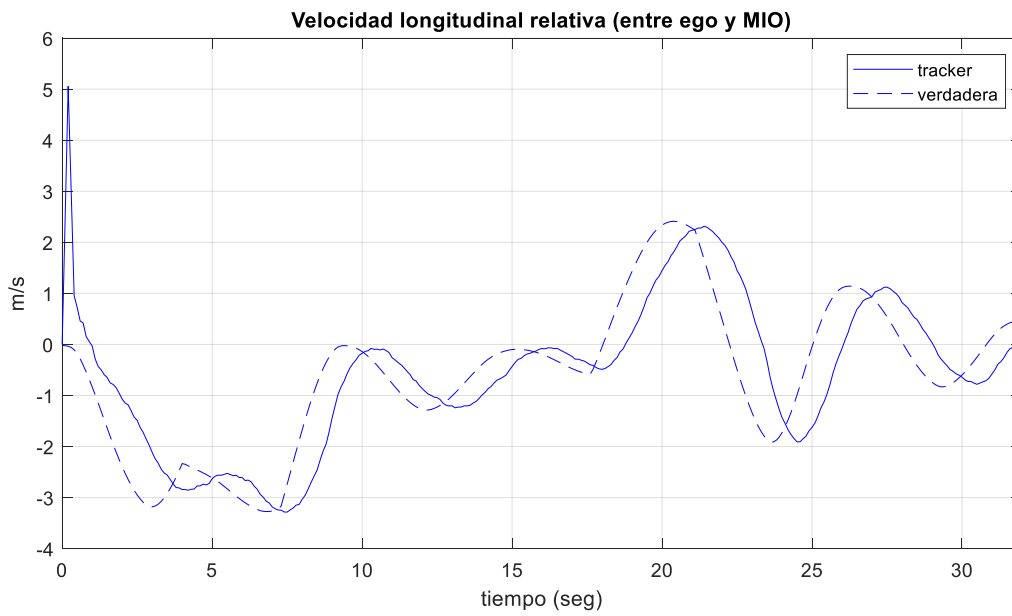
Continuando con nuestro análisis, de la Figura 8.2.3.3 podemos observar el comportamiento del controlador SMC ante este escenario, en general se presenta un comportamiento muy bueno, durante el primer cambio de velocidad el vehículo EGO se mantiene a una distancia segura; sin embargo, en el segundo cambio, se pierde por muy poco esta referencia y se tiene un porcentaje de oscilación alrededor de la distancia segura. En definitiva, este controlador sigue de manera adecuada los cambios de velocidad y la distancia requerida para evitar una colisión entre los vehículos, no obstante, el mismo comportamiento del controlador hace que la señal de aceleración del controlador cambie en magnitudes grandes en intervalos de tiempo muy pequeños, lo cual como se ha mencionado anteriormente, no es adecuado para el motor del vehículo.

En el caso del controlador CTG, de la Figura 8.2.3.4 se observa que al igual que el controlador SMC mantiene una distancia por encima de la distancia segura, cruzando un poco este límite en el segundo cambio de velocidad. Hay un pequeño retraso entre la velocidad del vehículo EGO y el MIO cuando se presentan los cambios de velocidad y en ambos casos la velocidad sigue la referencia adecuadamente, cabe señalar que con el cambio de velocidad requiere una mayor cantidad de tiempo para auto ajustarse. La señal de aceleración proporcionada por este controlador presenta un comportamiento suave en donde se presentan cambios de aceleración adecuados.

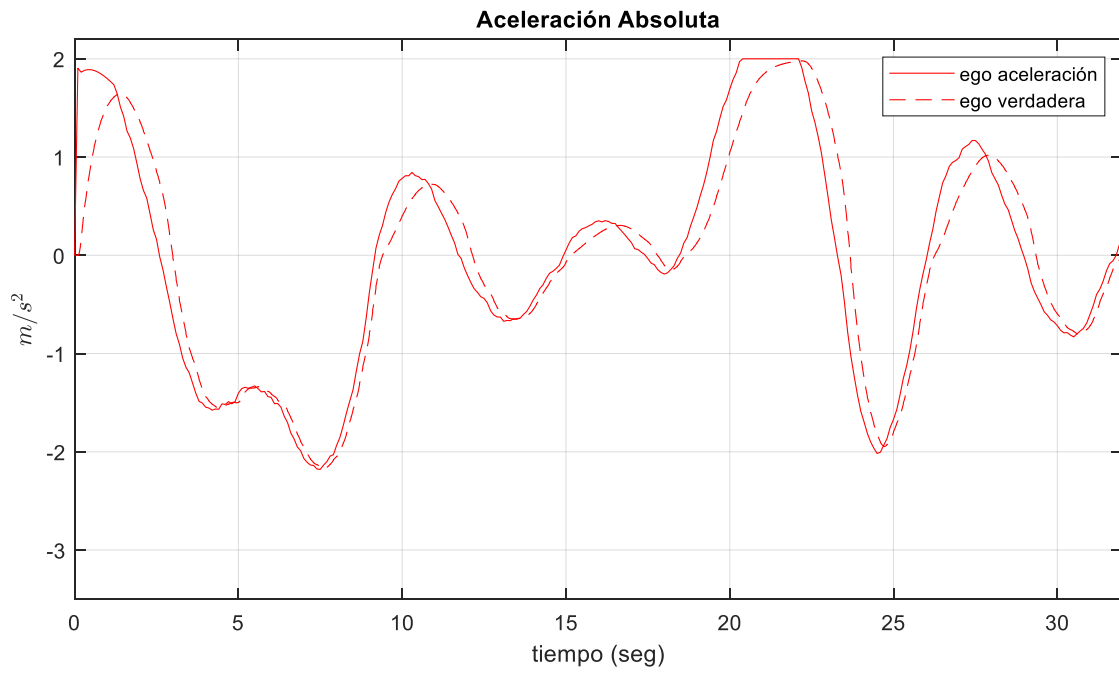
a)



b)



c)



d)

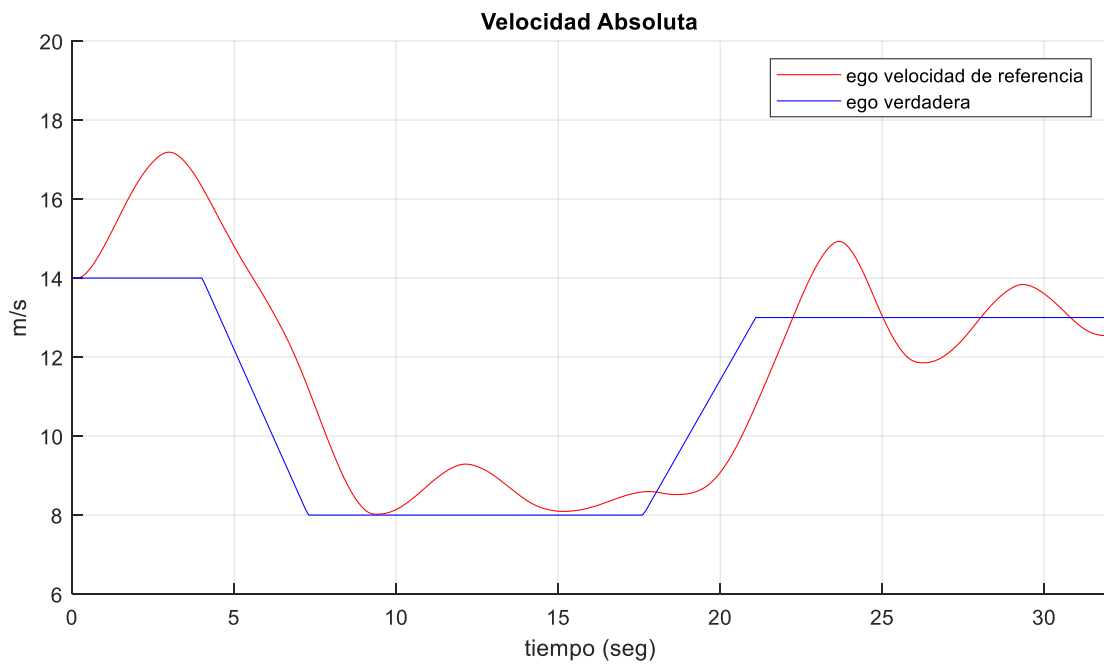
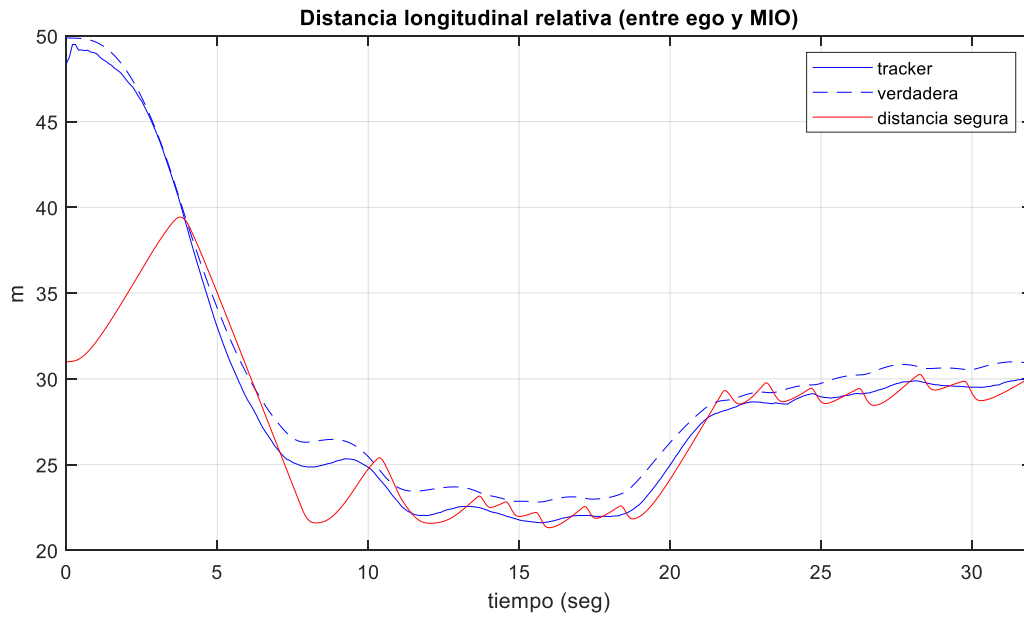


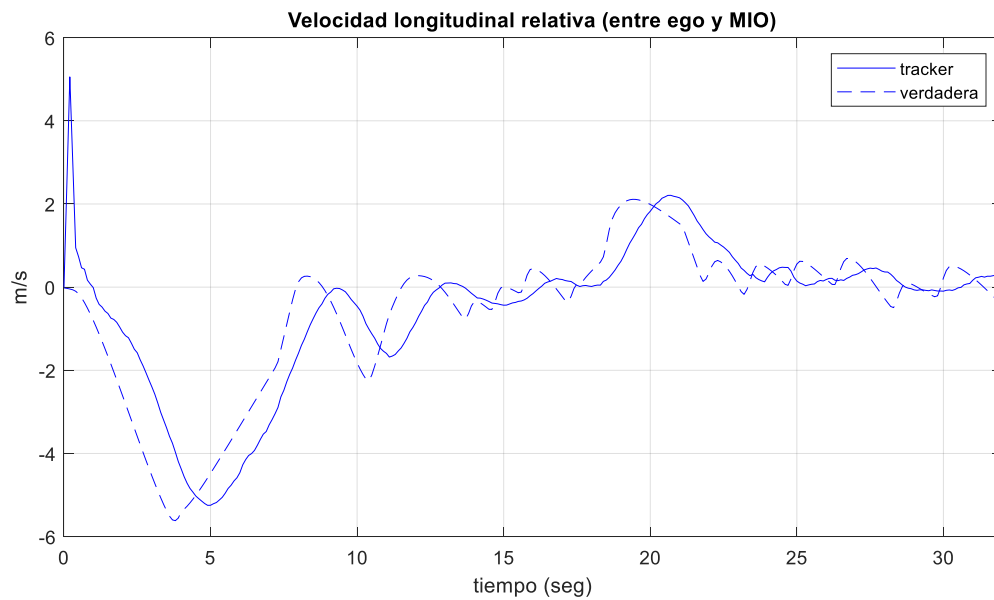
Figura 8.2.3.4. Desempeño del controlador CTG de ACC, a) Distancia longitudinal relativa, b) velocidad longitudinal relativa, c) Aceleración absoluta, d) Velocidad absoluta

Siguiendo la descripción anterior, el desempeño del controlador MPC con horizonte de predicción (Hp) de 20 y 40 se puede observar en las Figuras 8.2.3.5 y 8.2.3.6 respectivamente.

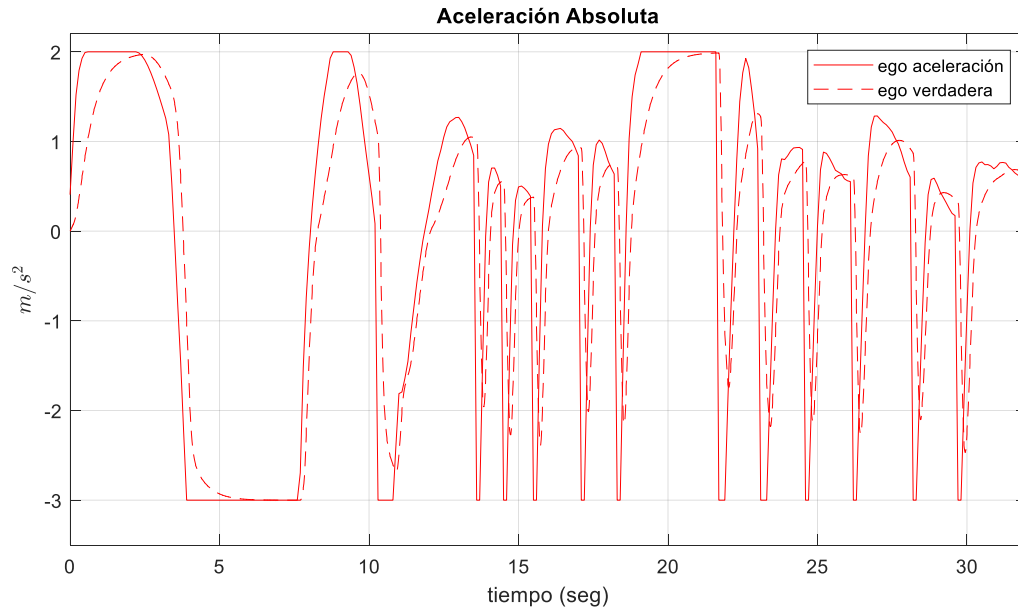
a)



b)



c)



d)

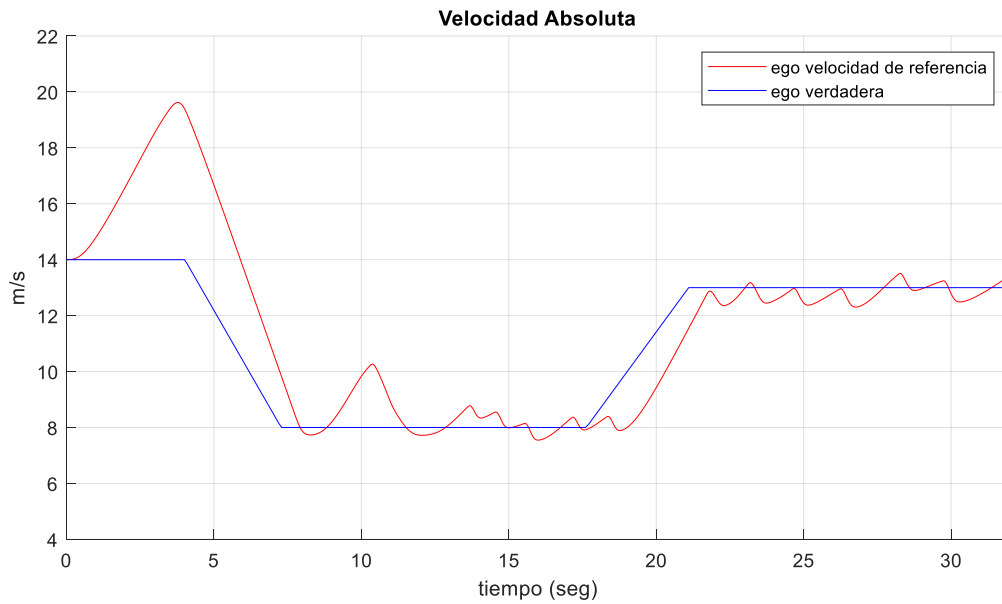
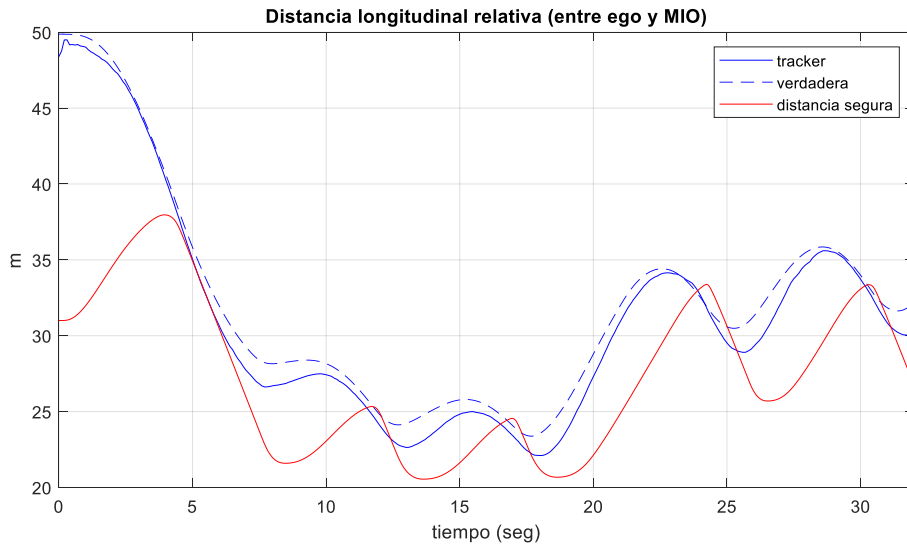


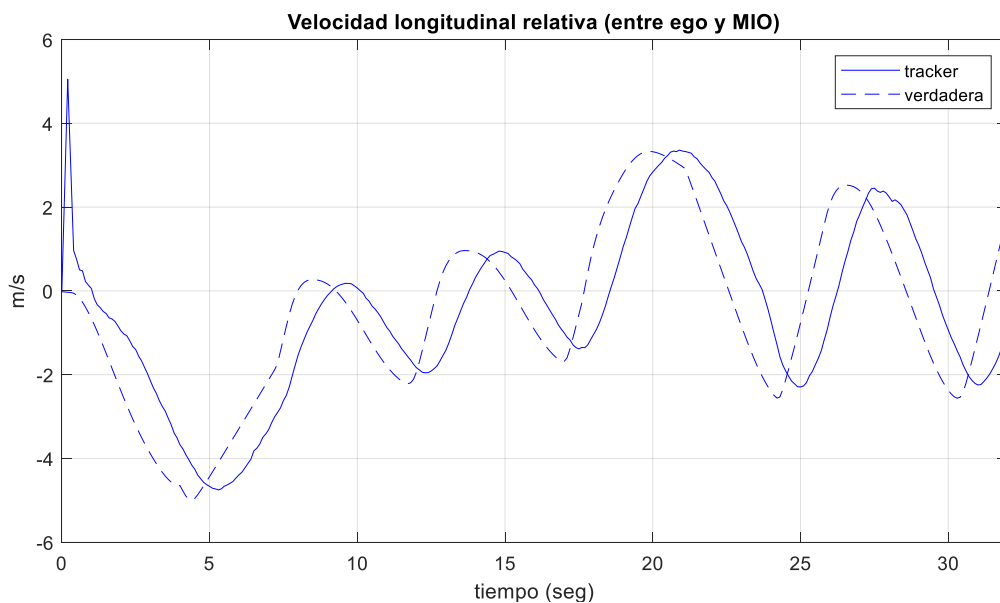
Figura 8.2.3.5. Desempeño del controlador MPC Hp de 20 del ACC, a) Distancia longitudinal relativa, b) velocidad longitudinal relativa, c) Aceleración absoluta, d) Velocidad absoluta

En el caso del controlador con Hp de 20, de la Figura 8.2.3.5 se observa que durante la mayoría de tiempo de simulación la distancia se mantiene oscilando con valores pequeños en la distancia segura, esto se comprueba una vez que observamos la gráfica de velocidad. En ella se visualiza que solo el primer cambio de velocidad tiene una oscilación grande, que en cuestión de segundos se estabiliza, mientras que el segundo cambio tiene un pequeño retraso y la velocidad del vehículo no presenta un sobrepaso considerable de la velocidad de referencia. Con respecto a la aceleración se sigue presentando un cambio de aceleración rápido; sin embargo, no al grado del controlador SMC.

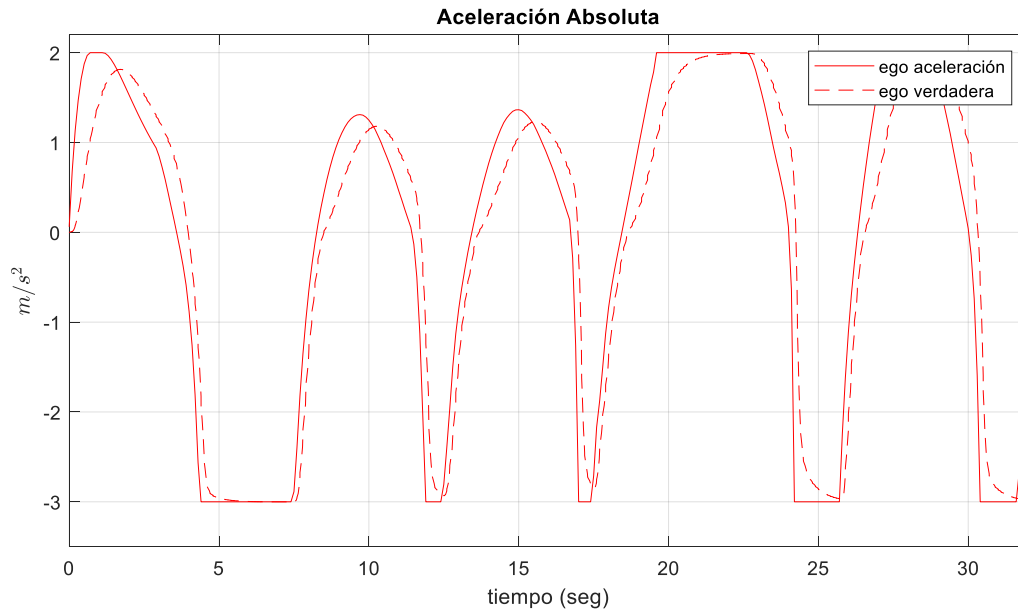
a)



b)



c)



d)

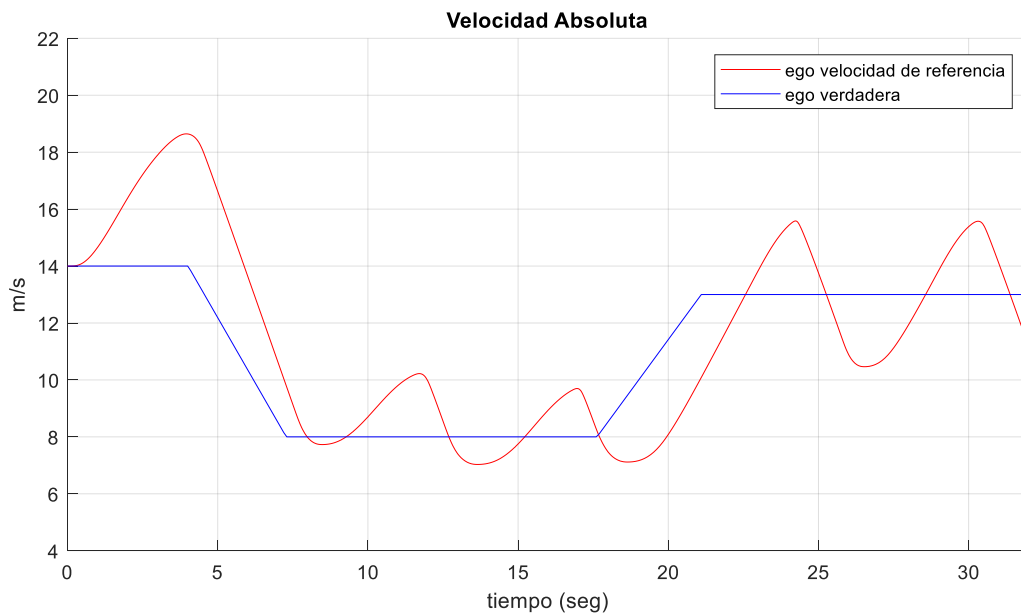


Figura 8.2.3.6. Desempeño del controlador MPC Hp de 40 del ACC, a) Distancia longitudinal relativa, b) velocidad longitudinal relativa, c) Aceleración absoluta, d) Velocidad absoluta

De la Figura 8.2.3.6 para un Hp de 40, la principal diferencia apreciable es la señal de aceleración, si bien el comportamiento es el mismo, este sucede en intervalos de tiempo mayores, por lo cual ocasiona que al momento de seguir la velocidad se presenten oscilaciones de mayor

amplitud. De lo anterior podemos concluir que para este tipo de cambios es más factible utilizar un Hp menor.

Ahora bien, para el escenario 03 se tomaron dos secciones de interés, las cuales representan los cambios de velocidad mencionados, de nuestra Figura 8.2.3.2 se obtiene el error RMSE de cada una, tal y como se muestra en la Figura 8.2.3.7 y 8.2.3.8, junto con la información en la Tabla 8.2.3.1 y 8.2.3.3 respectivamente.

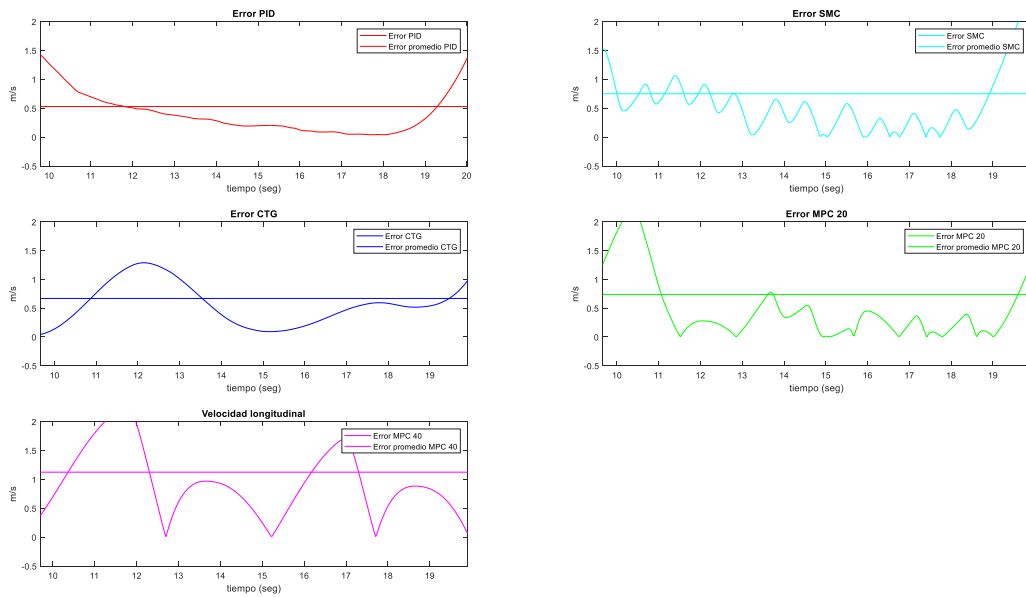


Figura 8.2.3.7. Variación del error RMSE de cada controlador con respecto a la velocidad de referencia 1 de 8 m/s.

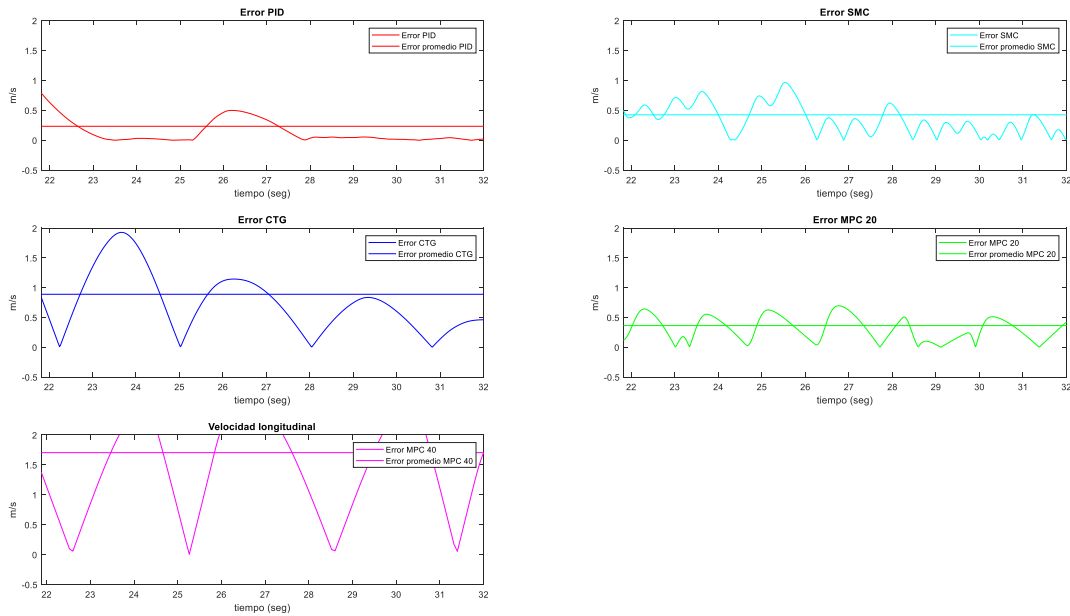


Figura 8.2.3.8. Variación del error RMSE de cada controlador con respecto a la velocidad de referencia 2 de 13 m/s.



ESCENARIO 3_08								
Controlador	Vmax	%V	t1	$\tau$	te	RMSE	%RMSE vel	
Clásico PID	0	0	18	7.54	15	0.530	6.63	
SMC	0	0	14.8	6.84	13.24	0.7515	9.39	
CTG	0	0	9.42	7.10	15.32	0.6698	8.37	
MPC 20	7.73	3.375	8	6.70	14.02	0.7389	9.24	
MPC 40	7.74	3.25	8	6.70	13.71	1.1250	14.06	

Tabla 8.2.3.1. Comparación del rendimiento de los controladores diseñados, escenario 02.

En la Tabla superior se visualiza que el menor error RMSE esta presente en los controladores CTG y MPC; sin embargo, es necesario realizar una ponderación para concluir cual es el controlador que presenta el mejor comportamiento, por tal motivo se realizó la matriz de decisión presentada en la Tabla 8.2.3.2. De igual forma, cabe recalcar que en el caso articular del controlador CTG no presenta sobre pico debido a que nunca llega en su totalidad a la velocidad de referencia, se aproxima casi en su totalidad en el segundo 9.42, mientras tanto, el SMC oscila en la referencia siguiendo su comportamiento característico sin presentar un sobrepaso como tal.

SELECCIÓN DE CONTROLADOR									
Controlador	Vmax	%V	t1	$\tau$	te	RMSE	%RMS E vel	Puntuación total	Orden de Desempeño
<b>Ponderación (0-5)</b>	1	1	3	3	5	5	5		
Clásico PID	0.00	0.00	54.00	22.62	75.00	2.65	33.13	187.40	4
SMC	0.00	0.00	44.40	20.52	66.20	3.76	46.97	181.85	3
CTG	0.00	0.00	28.26	21.30	76.60	3.35	41.86	171.37	2
MPC 20	7.73	3.38	24.00	20.10	70.10	3.69	46.18	164.08	1
MPC 40	7.74	3.25	24.00	20.10	68.55	5.63	70.31	188.59	5

Tabla 8.2.3.2. Matriz de decisión para seleccionar el controlador con el mejor desempeño, escenario 03 primera referencia.

De la Tabla 8.2.3.2, se concluye que el mejor comportamiento en este primer cambio de referencia lo presenta el controlador MPC con un Hp de 20, seguido del CTG y SMC por una diferencia de 7 y 10 puntos respectivamente. En realidad, la verdadera diferencia recae en el tiempo que requiere cada controlador para alcanzar la estabilidad en la oscilación de la señal y el tiempo en cruzar la referencia por primera vez, esto debido a que el MPC tiene una respuesta 4 y

20 segundos más rápida que el CTG y SMC respectivamente, cabe resaltar que la transición más suave se presenta en el controlador PID, con un comportamiento similar al resto de controladores.

ESCENARIO 3_13								
Controlador	Vmax	%V	t1	T	te	RMSE	%RMSE vel	
Clásico PID	13.007	0.0538	24.92	21.10	23.61	0.234	1.80	
SMC	13.01	0.076	24.34	20.74	26.26	0.4256	3.27	
CTG	14.92	14.76	22.26	21.30	25.06	0.8898	6.84	
MPC 20	13.17	1.30	22.98	20.90	21.92	0.3638	2.80	
MPC 40	15.57	19.76	22.6	21.61	22.52	1.7045	13.11	

Tabla 8.2.3.3. Comparación del rendimiento de los controladores diseñados, escenario 03 segunda referencia.

Continuando con el análisis del segundo cambio de velocidad, de la Tabla 8.2.3.3 se visualiza que la mayoría de los controladores tienen un error menor con respecto a la segunda velocidad de referencia de 13 [m/s]. De igual forma tienen tiempos de respuesta similares; sin embargo, donde se observa la mayor diferencia es en el sobre pico inicial que tienen, siendo el PID, SMC y el MPC 20 los que presentan un menor sobrepaso.

Siguiendo el mismo procedimiento que con los controladores anteriores, se realizó una matriz de decisión, la cual se puede observar en la Tabla 8.2.3.4.

SELECCIÓN DE CONTROLADOR									
Controlador	Vmax	%V	t1	$\tau$	te	RMSE	%RMSE vel	Puntuación total	Orden de desempeño
<b>Ponderación (0-5)</b>	1	1	3	3	5	5	5		
Clásico PID	13.01	0.05	74.76	63.30	118.05	1.17	9.00	266.28	1
SMC	13.01	0.08	73.02	62.22	131.30	2.13	16.37	298.12	3
CTG	14.92	14.77	66.78	63.90	125.30	4.45	34.22	324.34	4
MPC 20	13.17	1.31	68.94	62.70	109.60	1.82	13.99	271.53	2
MPC 40	15.57	19.77	67.80	64.83	112.60	8.52	65.56	354.65	5

Tabla 8.2.3.4. Matriz de decisión para seleccionar el controlador con el mejor desempeño, escenario 03 primera referencia.

De la Tabla 8.2.3.4, se concluye que el controlador PID tiene el mejor comportamiento seguido del MPC 20, esto se debe a que se estabilizaron en un tiempo menor con una oscilación baja con respecto a la referencia. De igual forma el SMC tiene un comportamiento bastante bueno; sin embargo, converge al valor deseado con mayor lentitud.

Del análisis anterior, se observa como el controlador MPC con Hp de 20 y 40 mantienen un desempeño consistente ante ambas perturbaciones, siendo que cuando se tiene un Hp de 20 el comportamiento destaca entre ellos. Por otro lado, el controlador CTG y SMC, mantienen un buen desempeño, aunque no se mantiene consistente ante las perturbaciones, principalmente ocasionado por el tiempo de respuesta de ambos controladores. Adicionalmente, el controlador PID mantiene la referencia con mayor lentitud, pero es compensado por su rápida acción ante la segunda perturbación.

### 8.2.4 Escenario 4

Recordando, este escenario presenta un cambio de carril proveniente de un vehículo en el carril adyacente, por lo que el vehículo EGO frena por un instante y retoma la velocidad puesta por el usuario del vehículo o la del vehículo MIO según corresponda, en la Figura 8.2.4.1 podemos observar el comportamiento de los controladores.

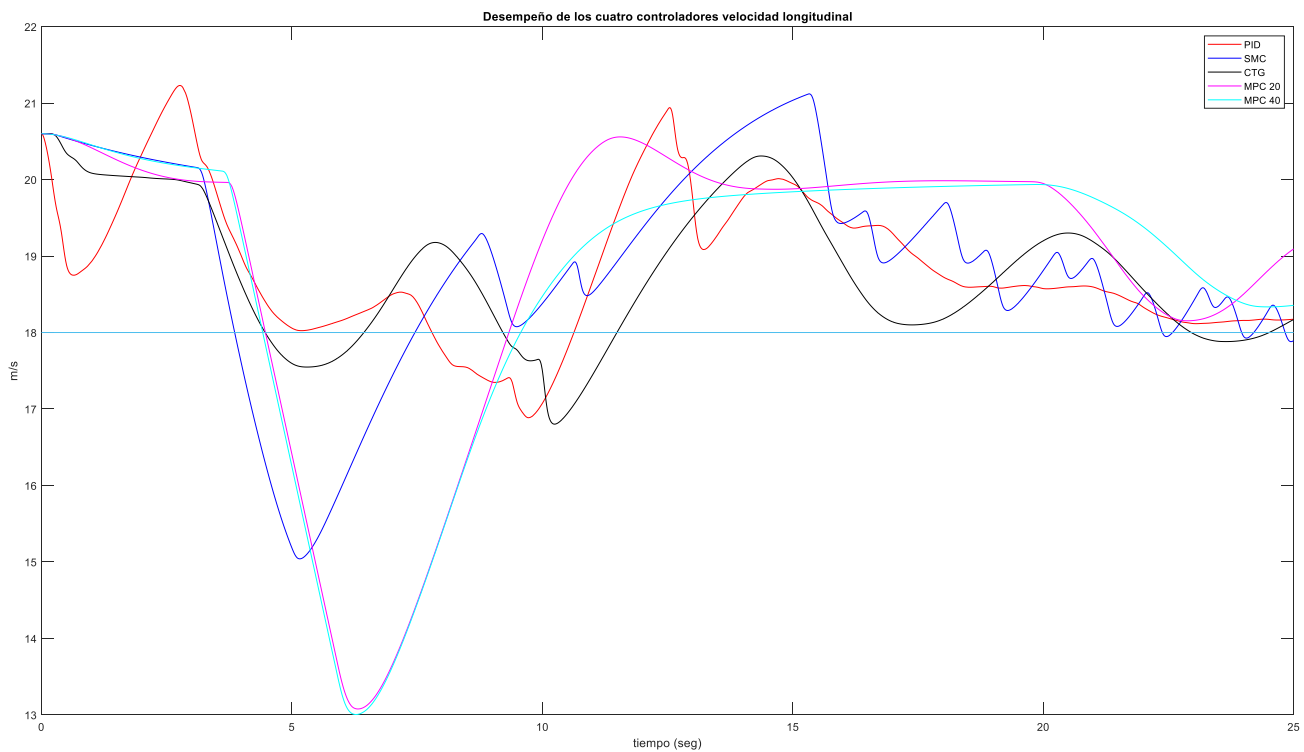
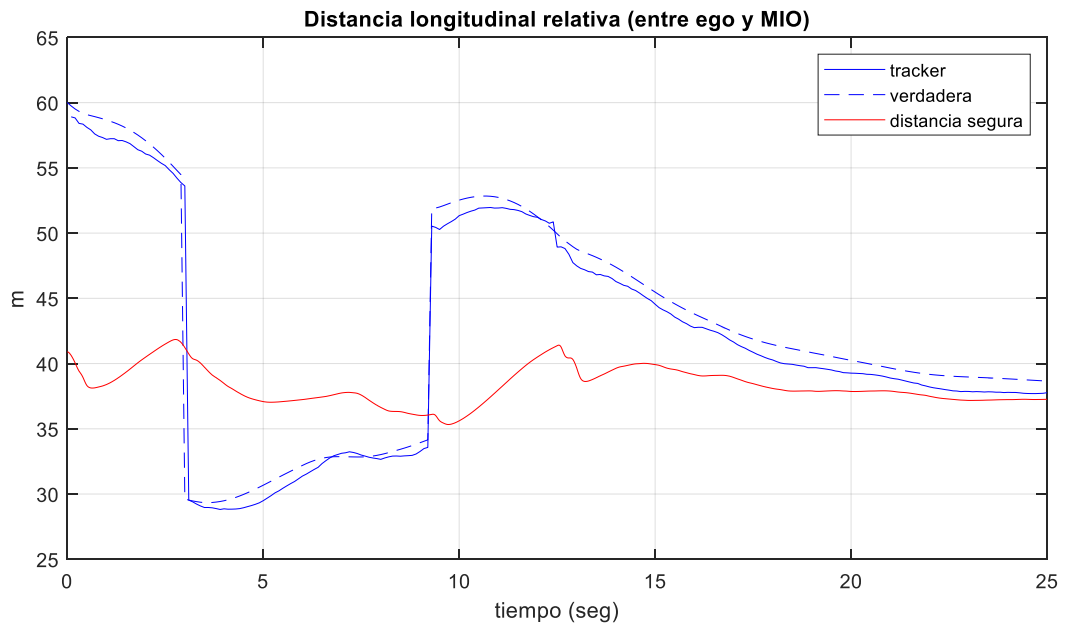


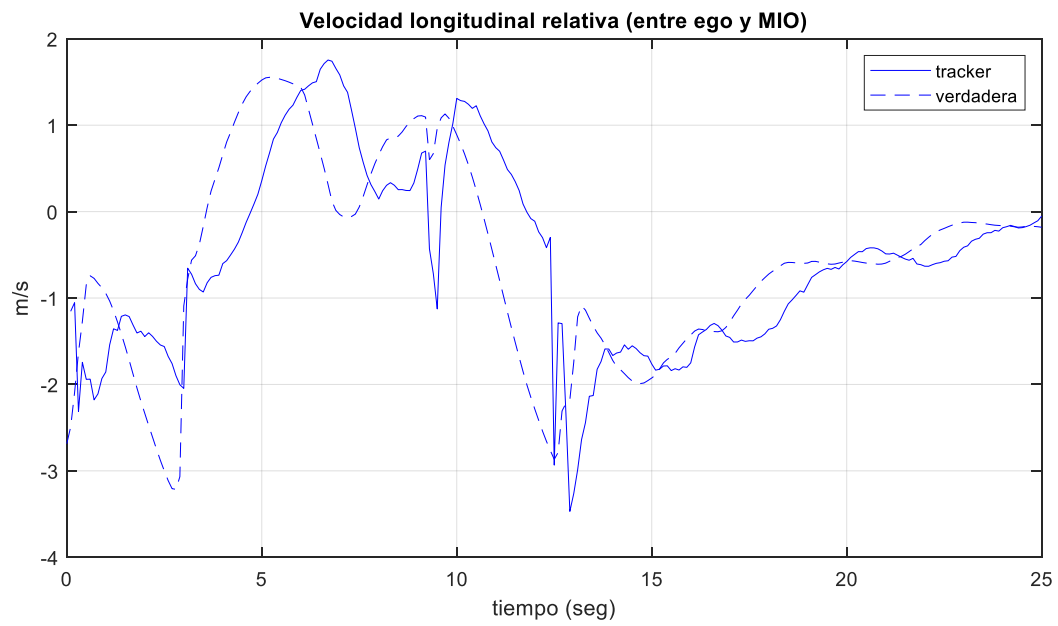
Figura 8.2.4.1. Escenario 04, comparación de los cuatro controladores y su efecto de la variación de la velocidad longitudinal del vehículo EGO.

Si verificamos el comportamiento de cada controlador por separado se obtiene lo siguiente:

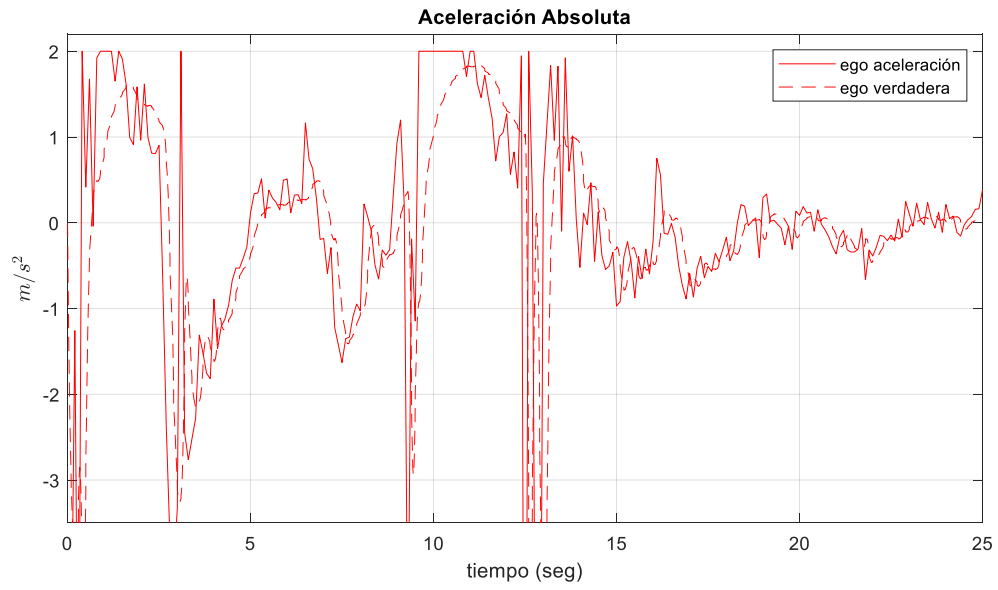
a)



b)



c)



d)

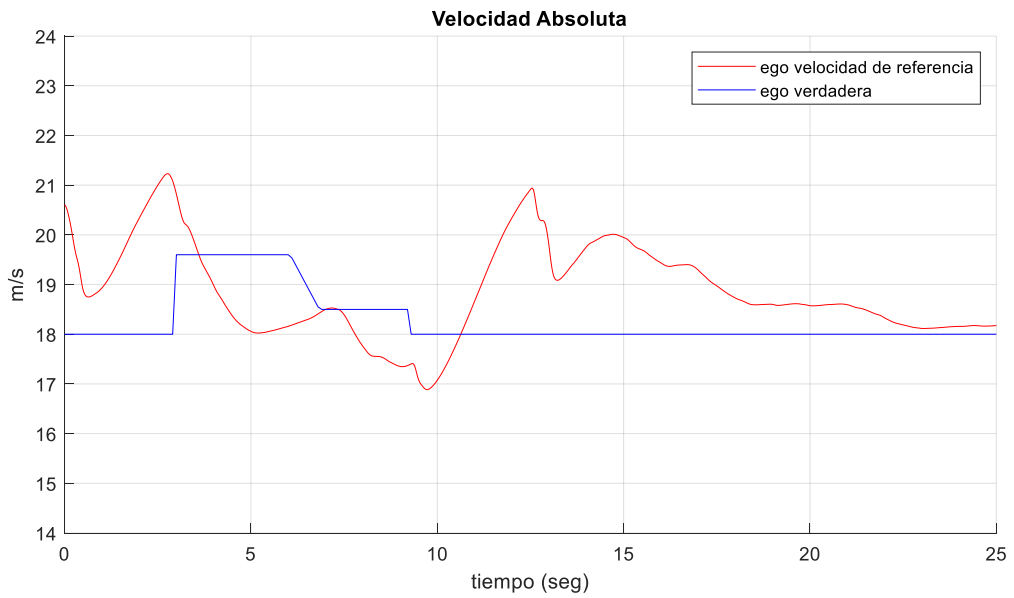
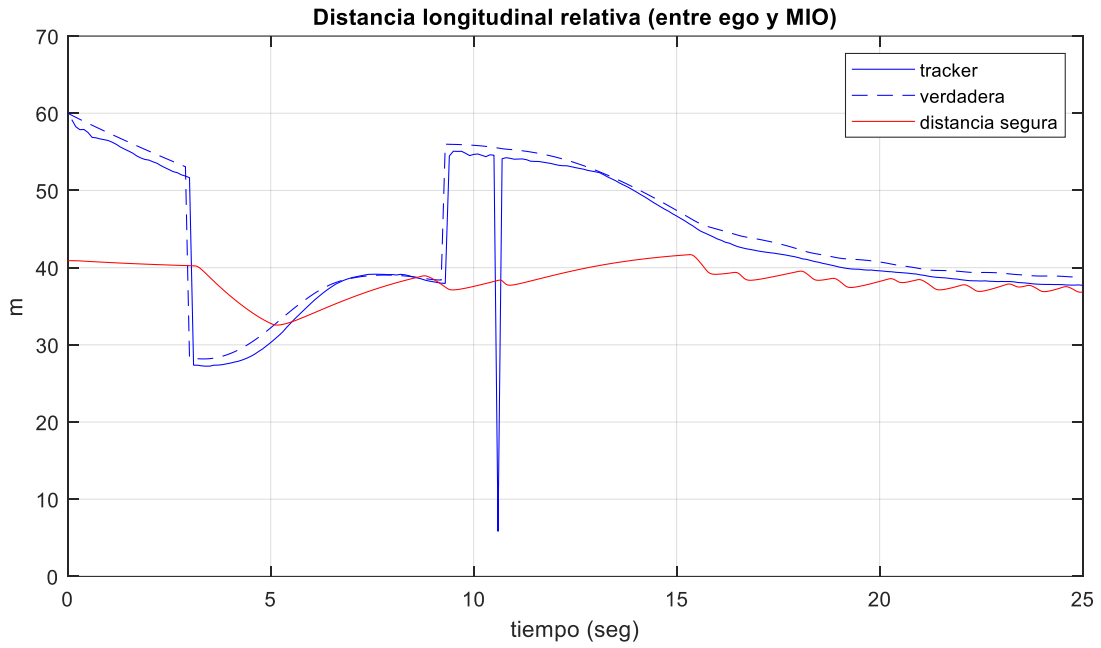


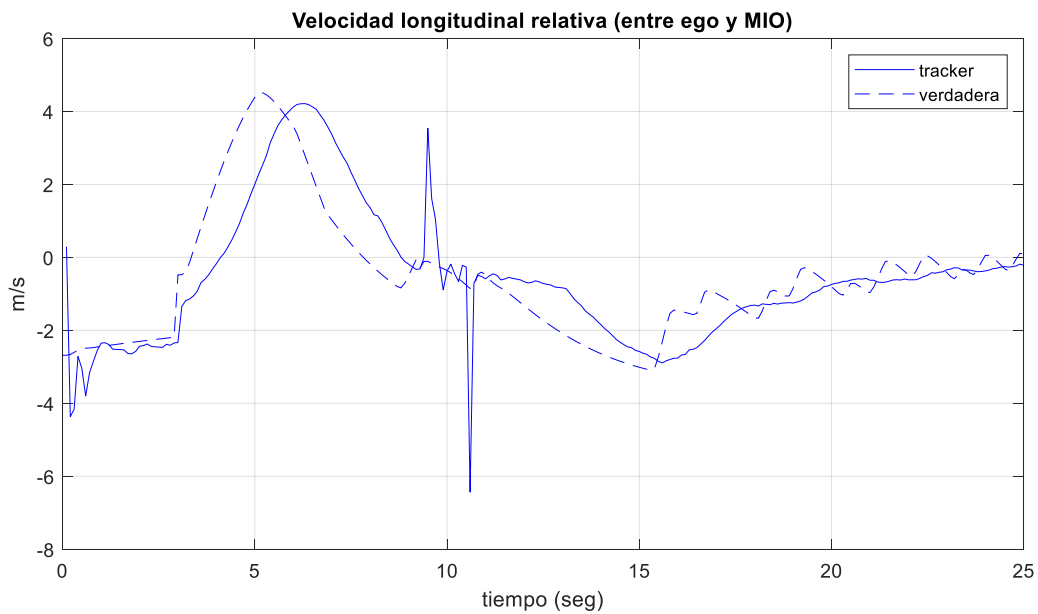
Figura 8.2.4.2. Desempeño del controlador PID de ACC, a) Distancia longitudinal relativa, b) velocidad longitudinal relativa, c) Aceleración absoluta, d) Velocidad absoluta

De la Figura 8.2.4.2, podemos observar que el controlador PID tiene una respuesta adecuada cuando el vehículo adyacente cambia de carril, al igual que el resto de los controladores logra seguir la velocidad de manera suave evitando una colisión con el vehículo en cuestión.

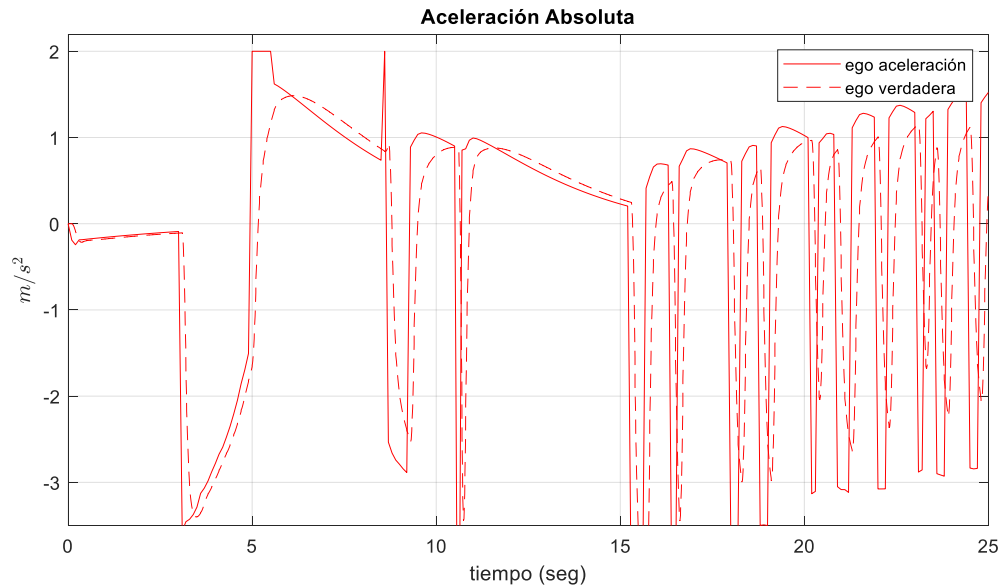
a)



b)



c)



d)

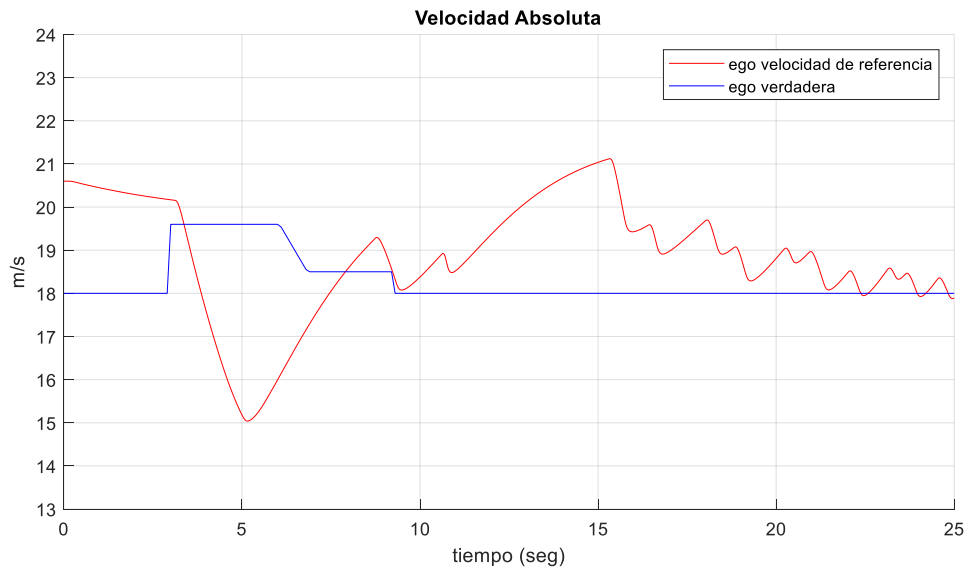
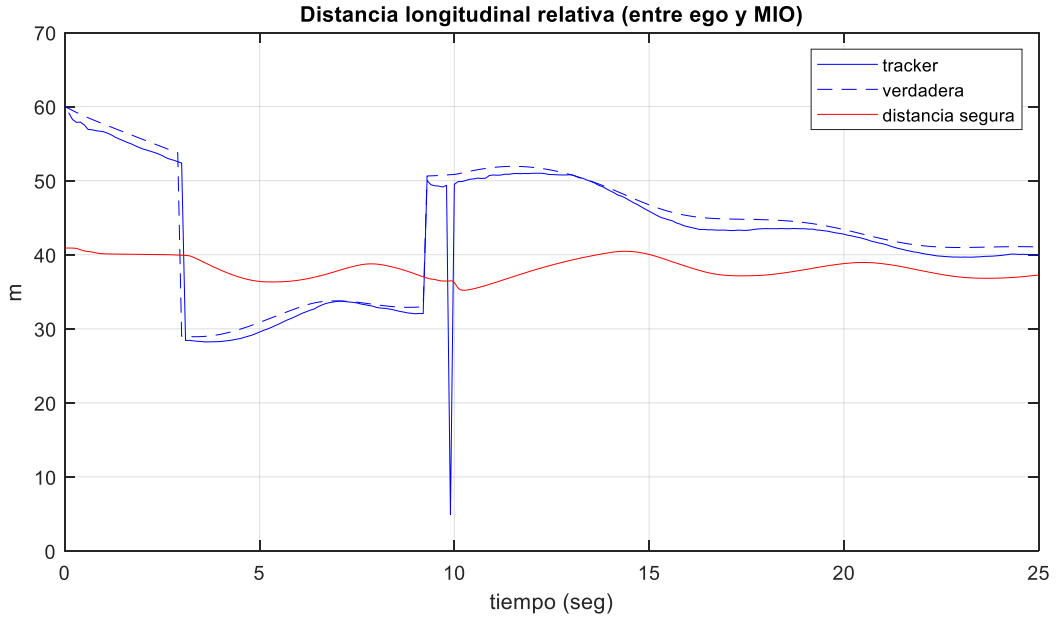


Figura 8.2.4.3. Desempeño del controlador SMC de ACC, a) Distancia longitudinal relativa, b) velocidad longitudinal relativa, c) Aceleración absoluta, d) Velocidad absoluta

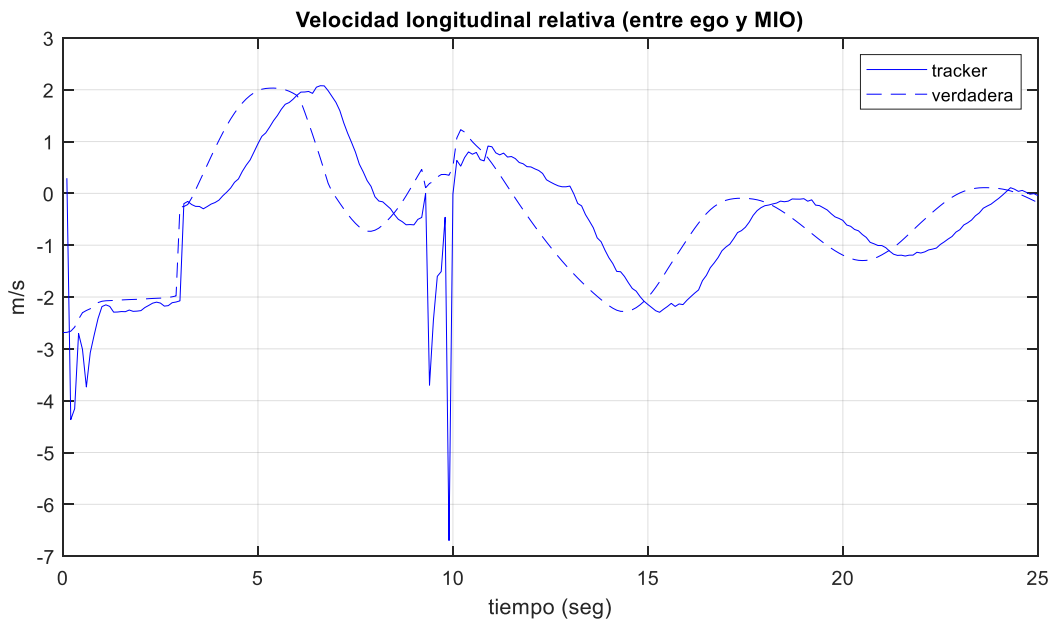
De la Figura 8.2.4.3, el controlador SMC realiza un frenado adecuado ante el primer acercamiento del vehículo adyacente, mantiene una distancia segura prácticamente durante toda la simulación, sin embargo, el comportamiento de su aceleración no es el más adecuada. También, se puede observar que antes de seguir la referencia de velocidad del vehículo MIO, intenta llegar a

la velocidad establecida por el usuario de 20 [m/s], es por eso que hay un incremento de velocidad mayor a la referencia y posteriormente esta disminuye a 18 [m/s].

a)

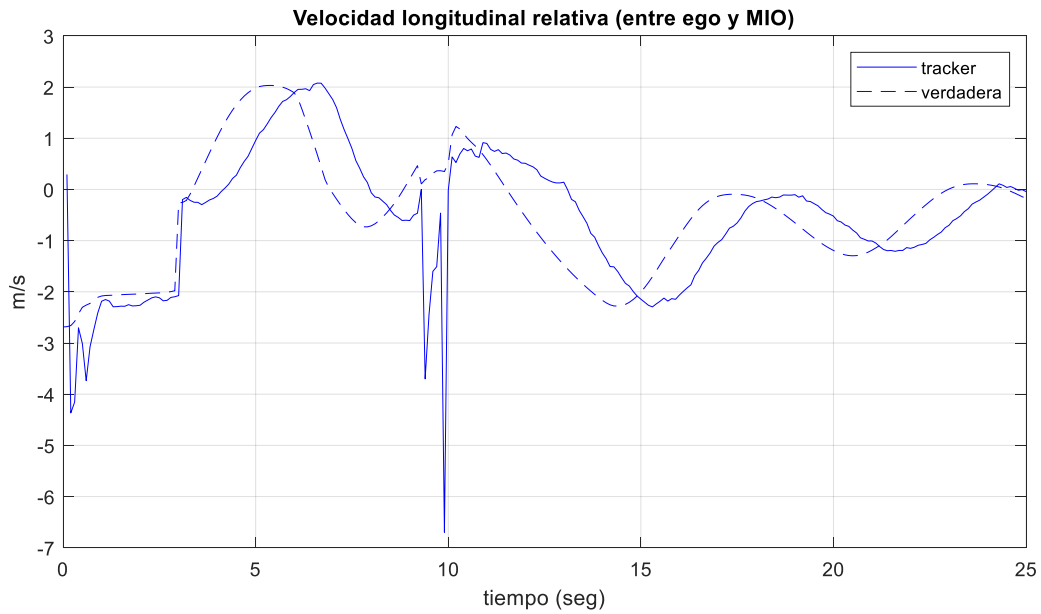


b)





c)



d)

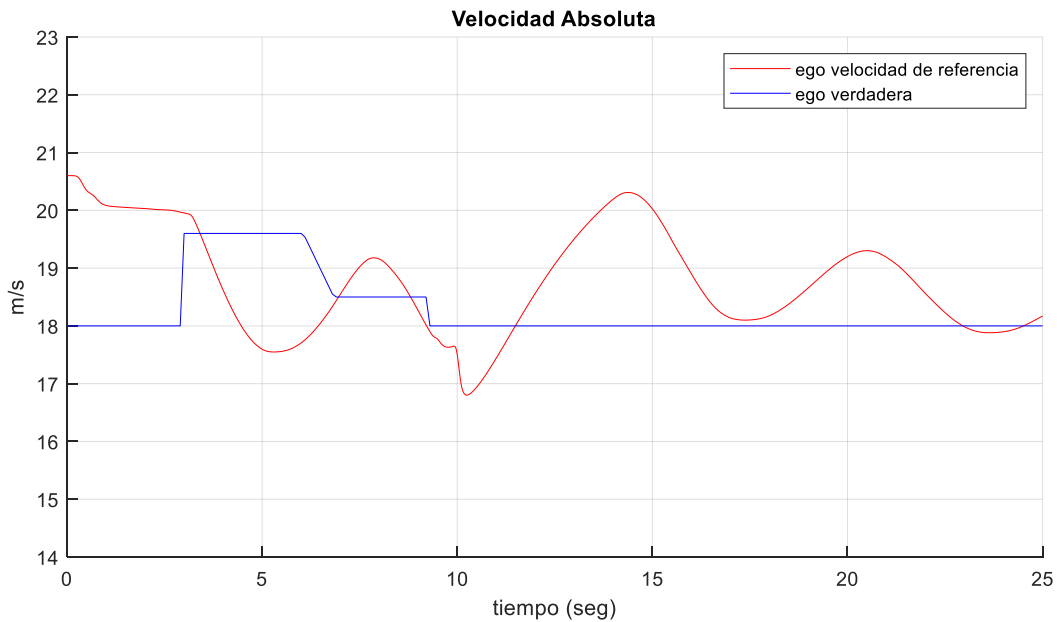
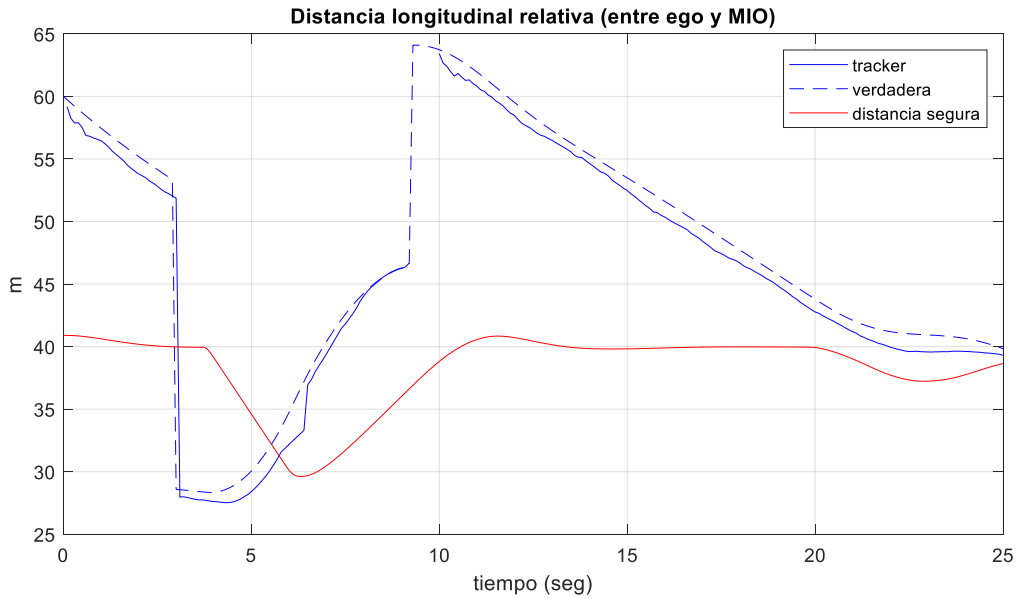


Figura 8.2.4.4. Desempeño del controlador CTG de ACC, a) Distancia longitudinal relativa, b) velocidad longitudinal relativa, c) Aceleración absoluta, d) Velocidad absoluta

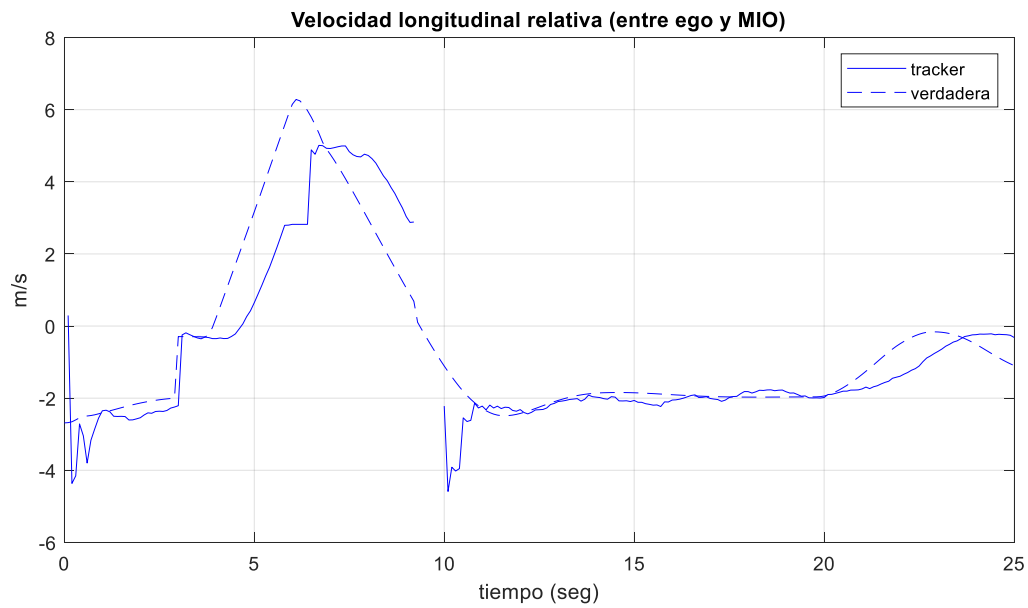
De la Figura 8.2.4.4, el controlador CTG realiza un frenado adecuado ante la perturbación y mantiene una distancia mayor a la distancia segura, se observa que tarda en ajustarse

nuevamente a la velocidad de referencia y tiene un pico no deseado al identificar la distancia entre ambos vehículos, ese mismo instante hace que la velocidad baje súbitamente, esto puede deberse al procesamiento de visión; sin embargo, nuevamente se ajusta, aunque con oscilaciones de amplitud grandes.

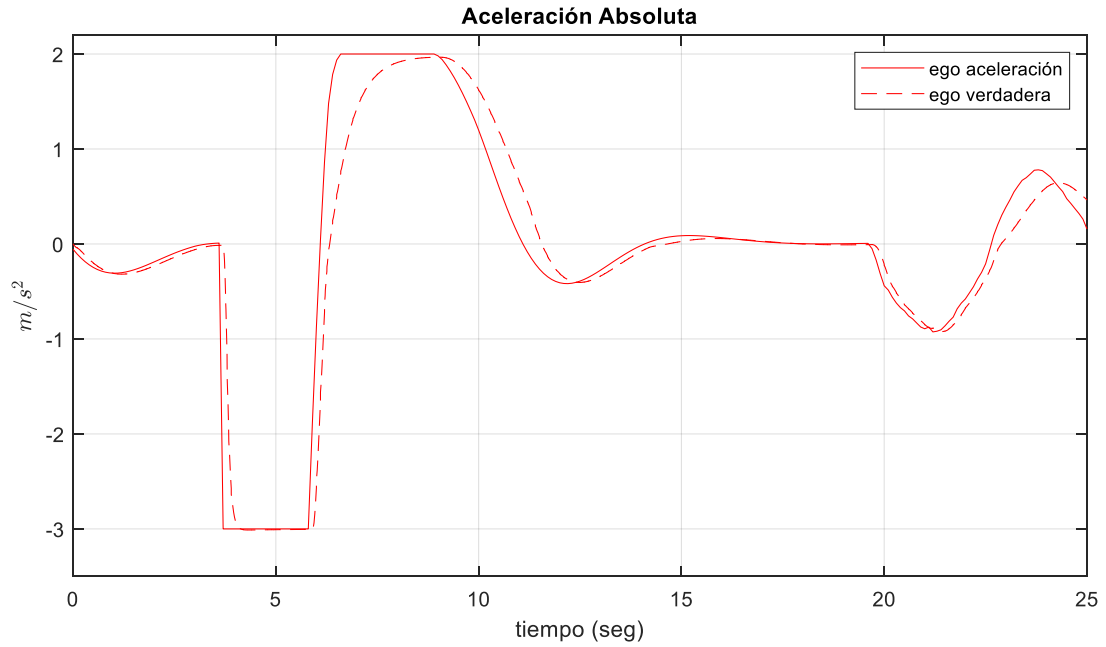
a)



b)



c)



d)

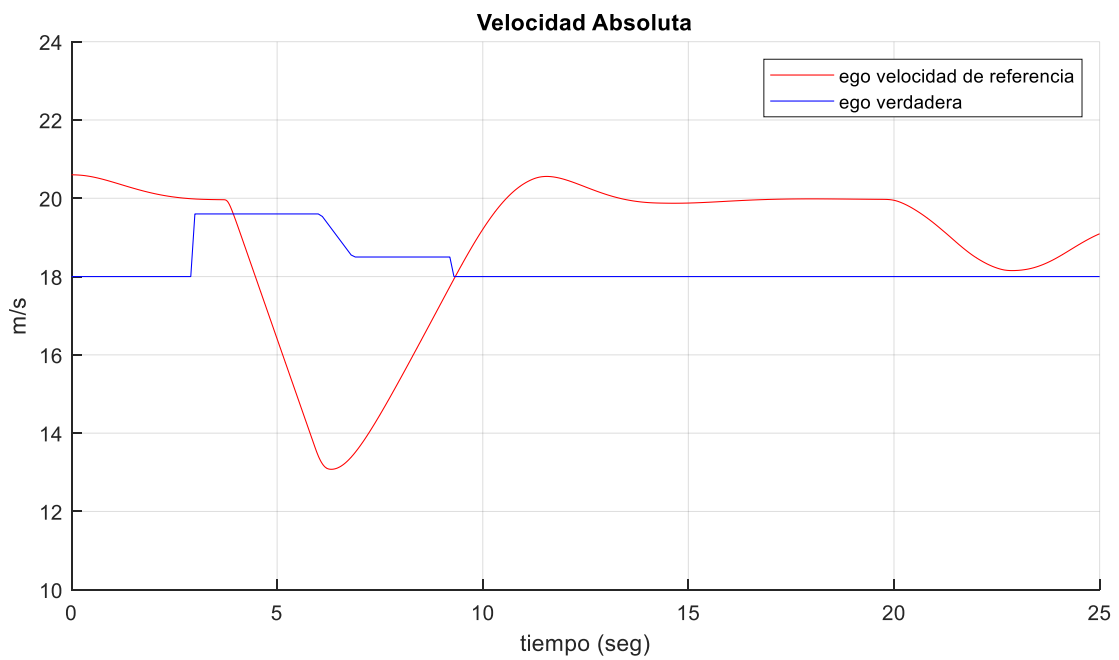
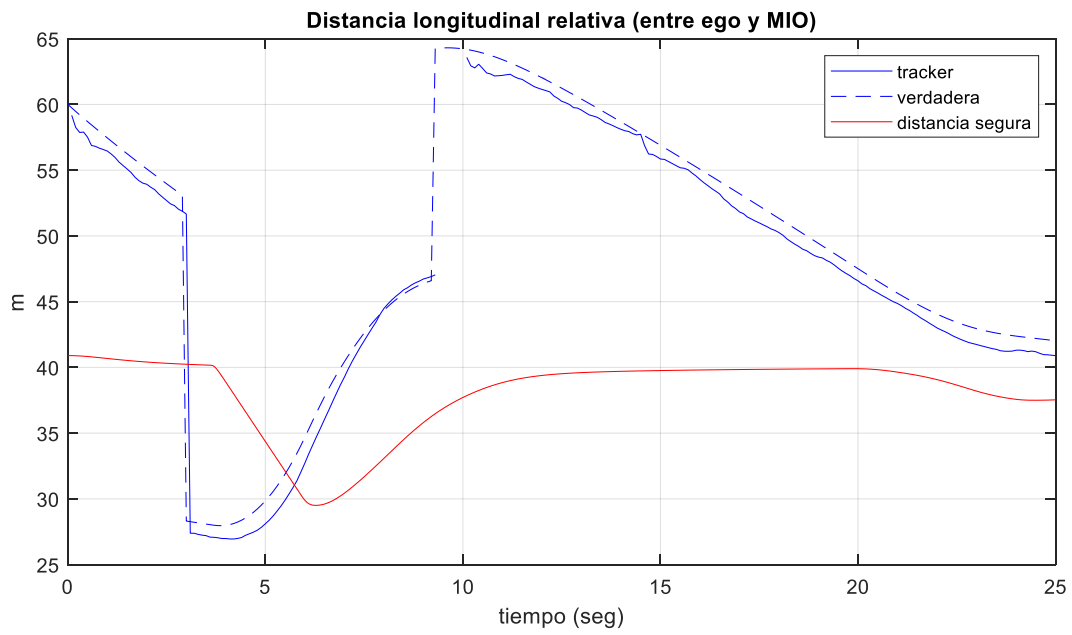
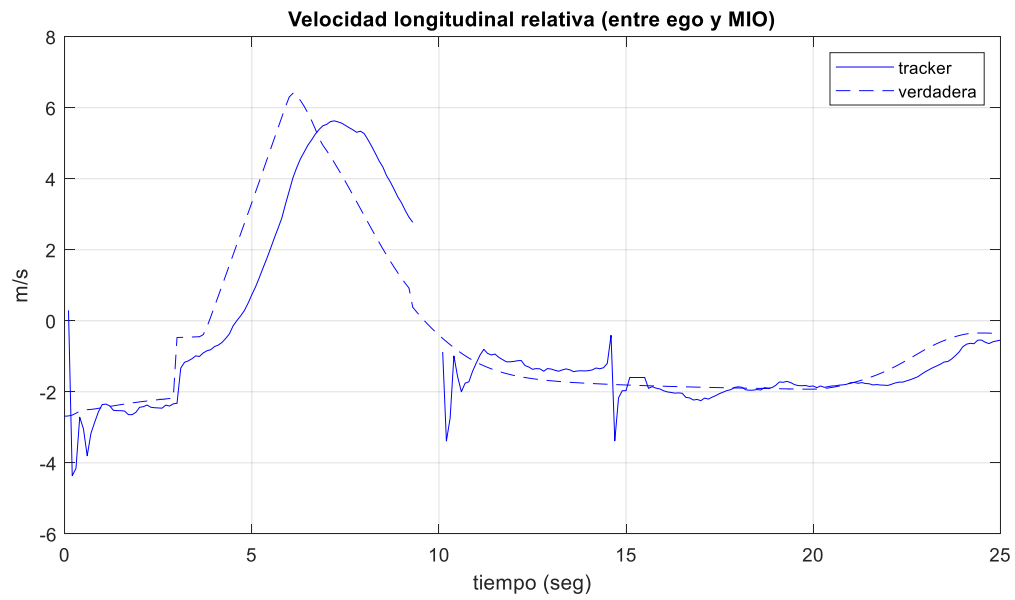


Figura 8.2.4.5. Desempeño del controlador MPC con HP de 20 del ACC, a) Distancia longitudinal relativa, b) velocidad longitudinal relativa, c) Aceleración absoluta, d) Velocidad absoluta

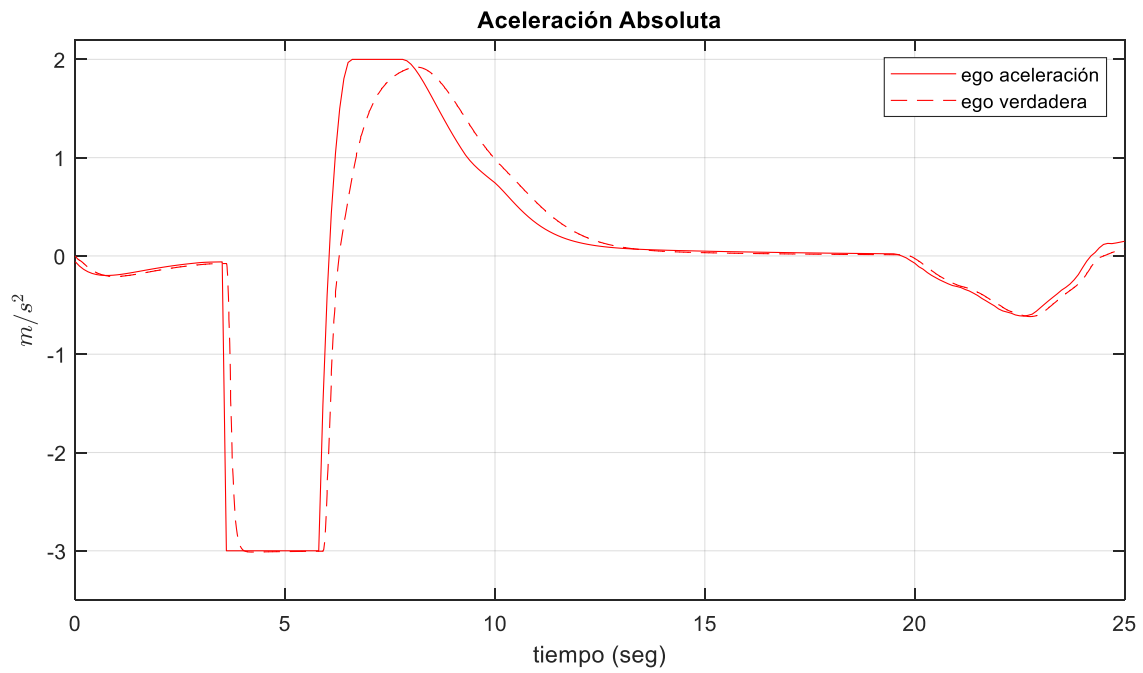
a)



b)



c)



d)

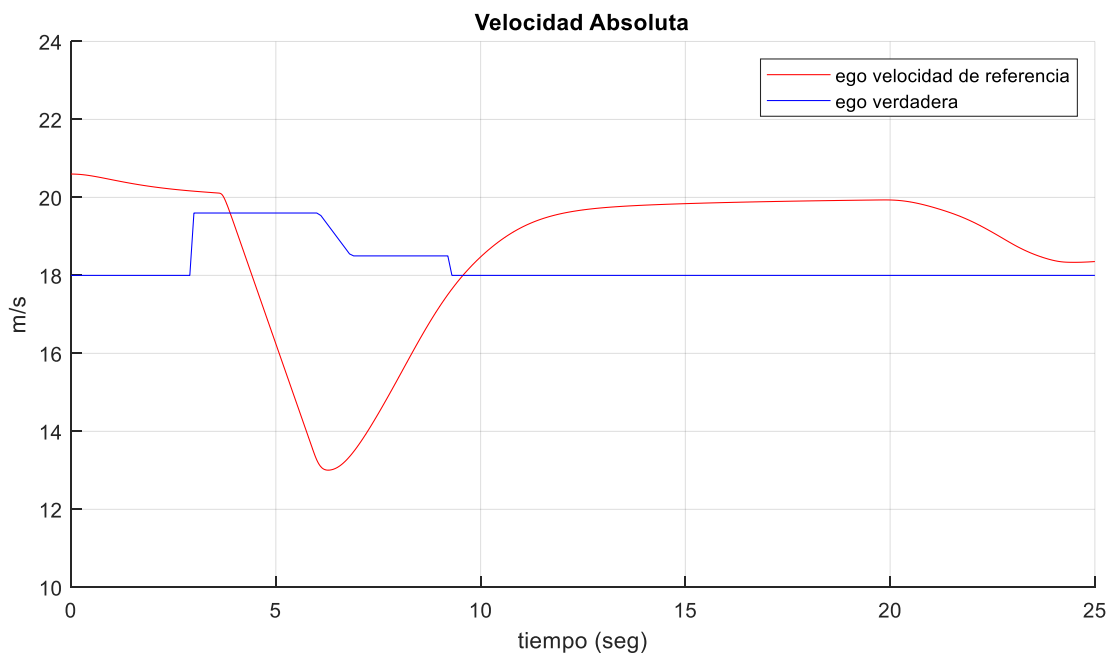


Figura 8.2.4.6. Desempeño del controlador MPC con HP de 40 del ACC, a) Distancia longitudinal relativa, b) velocidad longitudinal relativa, c) Aceleración absoluta, d) Velocidad absoluta

En conjunto la Figura 8.2.4.5 y 8.2.4.6, describen el comportamiento del controlador MPC cuando se tiene un Hp de 20 y 40 respectivamente. Al igual que los controladores anteriores, realiza un frenado adecuado justo cuando el vehículo adyacente cambia de carril; no obstante, es un frenado de mayor magnitud comparativamente hablando. Lo anterior ocasiona que se genere un espacio mayor entre los vehículos EGO y MIO, por lo cual el controlador MPC intenta en primera instancia llegar a la velocidad establecida por el usuario de 20 [m/s], viajando de esta forma casi por el resto de la simulación hasta que se encuentra nuevamente con el vehículo MIO y comienza reducir su velocidad al valor de referencia de 18 [m/s], esto sucede en ambos casos. En general ambos se comportan de una manera muy similar y la única diferencia visible es el sobre pico que tienen al alcanzar las referencias antes mencionadas.

Ahora bien, si calculamos el error RMSE de los controladores se obtiene la Figura 8.2.4.7, y tomando los puntos de interés para el análisis de este escenario se obtuvieron los resultados de la Tabla 8.2.4.1.

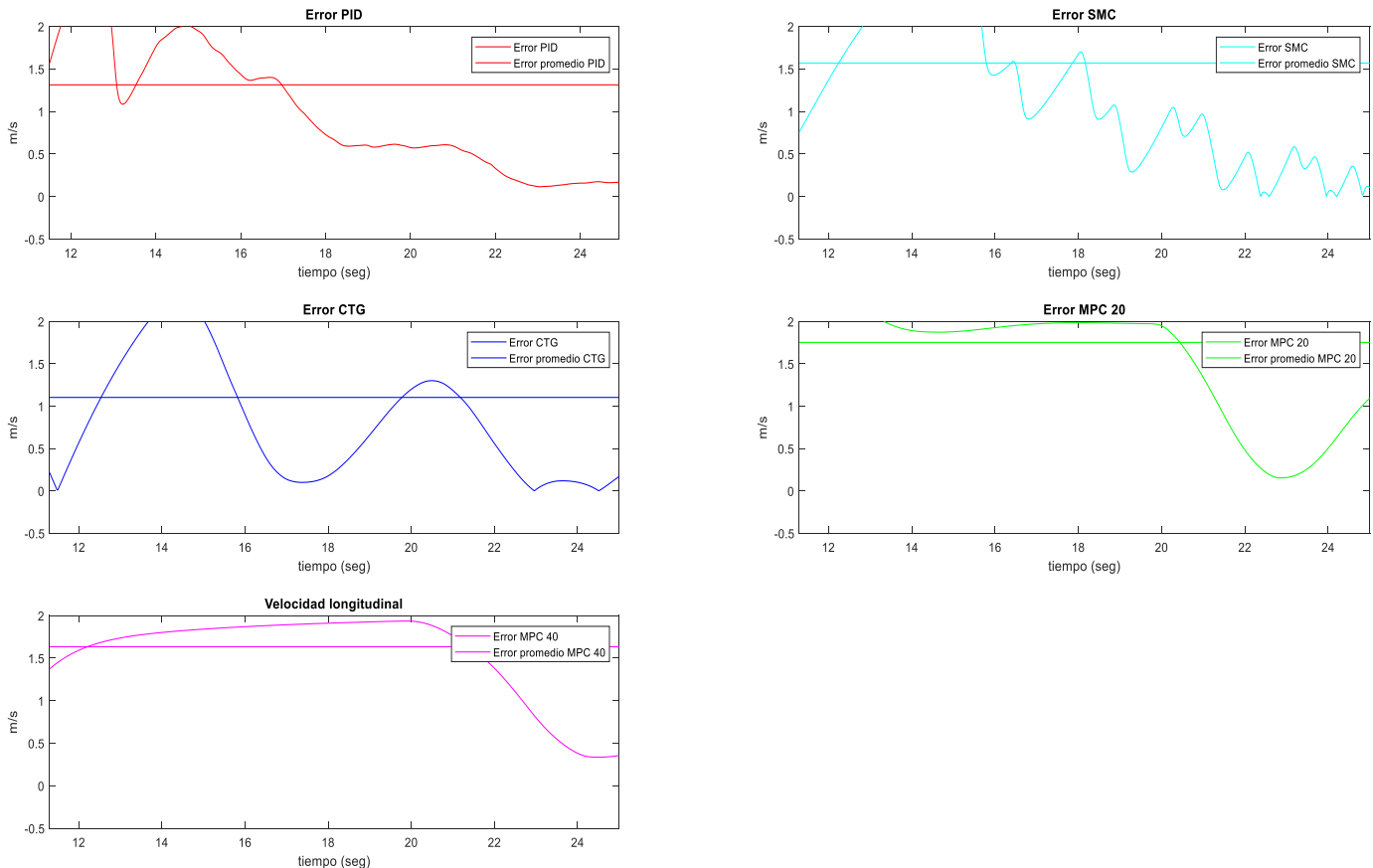


Figura 8.2.4.7. Variación del error RMSE de cada controlador con respecto a la velocidad de referencia

ESCENARIO 4									
Controlador	Vmax	%V	t1	$\tau$	te	RMSE	%RMSE vel	Tiempo de frenado	Velocidad en el tiempo de frenado
Clásico PID	20.93	16.28	10.619	9.94	23.2	1.310	7.28	11.22	16.89
SMC	21.12	17.33	9.48	6.66	23.24	1.5690	12.07	5.08	15.07
CTG	20.3	12.78	11.54	6.20	22.98	1.1032	8.49	5.28	17.54
MPC 20	20.55	2.75	20	9.14	17.34	0.8114	6.24	6.30	13.07
MPC 40	19.56	-2.2	19.92	9.14	19.98	0.6757	5.20	6.30	13.07

Tabla 8.2.4.1. Comparación del rendimiento de los controladores diseñados, escenario 04.

De igual forma que en los escenarios anteriores, se observa que el controlador PID hace un seguimiento paulatino de la señal y de manera similar, el resto de controladores tiene una disminución variable de velocidad en el primer instante de frenado, en el caso del SMC y CTG, este tiene una menor magnitud en comparación con el MPC, de igual forma los primeros dos frenan con un segundo de anticipación. En relación al error los cuatro controladores tienen valores diferentes, debido a que están siguiendo referencias diferentes; sin embargo, con respecto a cada una de ellas se determinó el porcentaje de oscilación para poder comparar los datos arrojados en la simulación. Lo anterior resulta en el siguiente índice de ponderación mostrado en la Tabla 8.2.4.2.

EVALUACIÓN DEL CONTROLADOR									
Controlador	Vmax	%V	t1	$\tau$	te	RMSE	%RMSE vel	Puntuación total	Orden de desempeño
<b>Ponderación (0-1)</b>	1	1	3	3	5	5	5		
Clásico PI	20.93	16.28	31.86	29.82	116.00	6.55	36.39	220.62	4
SMC	21.12	17.33	28.44	19.98	116.20	7.85	60.35	232.81	5
CTG	20.30	12.78	34.62	18.60	114.90	5.52	42.43	216.07	2
MPC 20	20.55	2.75	60.00	27.42	86.70	4.06	31.21	209.38	1
MPC 40	19.56	-2.20	59.76	27.42	99.90	3.38	25.99	216.45	3

Tabla 8.2.4.2. Índice de ponderación para seleccionar el controlador con el mejor desempeño, escenario 04.

De la Tabla 8.2.4.2, podemos observar que el MPC 20 tiene el mejor desempeño, aunque por muy poca diferencia con respecto al CTG y a su contraparte con Hp de 40. No obstante, cabe señalar que hubo una diferencia en el valor de referencia de los controladores debido a que la

lógica en el interruptor dedicado a realizar el cambio entre el controlador de velocidad y el de espaciamiento tomo valores diferentes en cada caso, generando una diferencia entre el controlador MPC y el resto de configuraciones de control.

### 8.2.5 Escenario 5

Tal y como se mencionó anteriormente, el escenario 5 se presenta un vehículo adyacente que cambia de carril; sin embargo, la principal diferencia es que este cambio ocurre en un lapso de tiempo menor, dicho comportamiento se ilustra en la Figura 8.2.5.1. Este análisis resulta útil para complementar el anteriormente descrito en el escenario 04, ya que contiene un distanciamiento menor entre los vehículos y la perturbación es mucho más rápida, esto quiere decir que la lógica del interruptor no afecta a la transición de los controladores, la cual como hemos mencionado resulta ser un cambio entre el control de cruceo y el control de cruceo adaptativo.

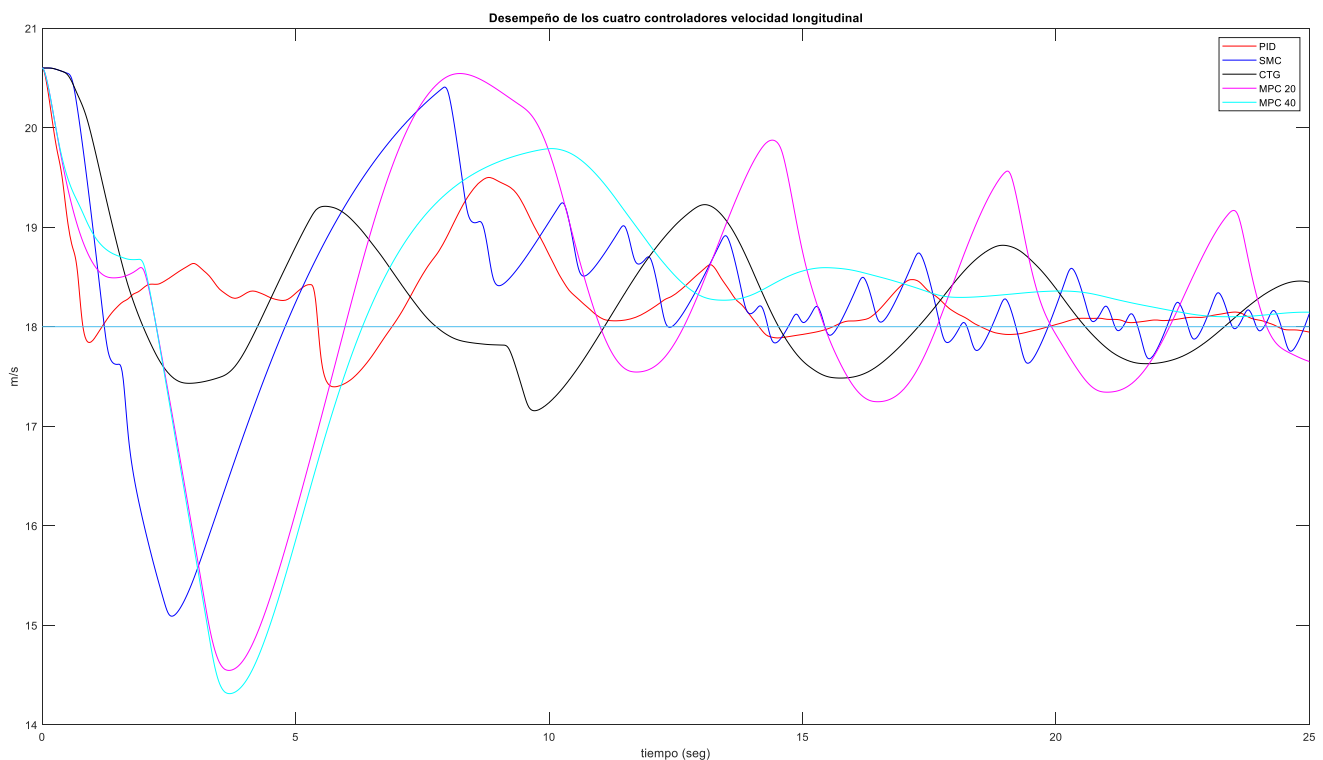
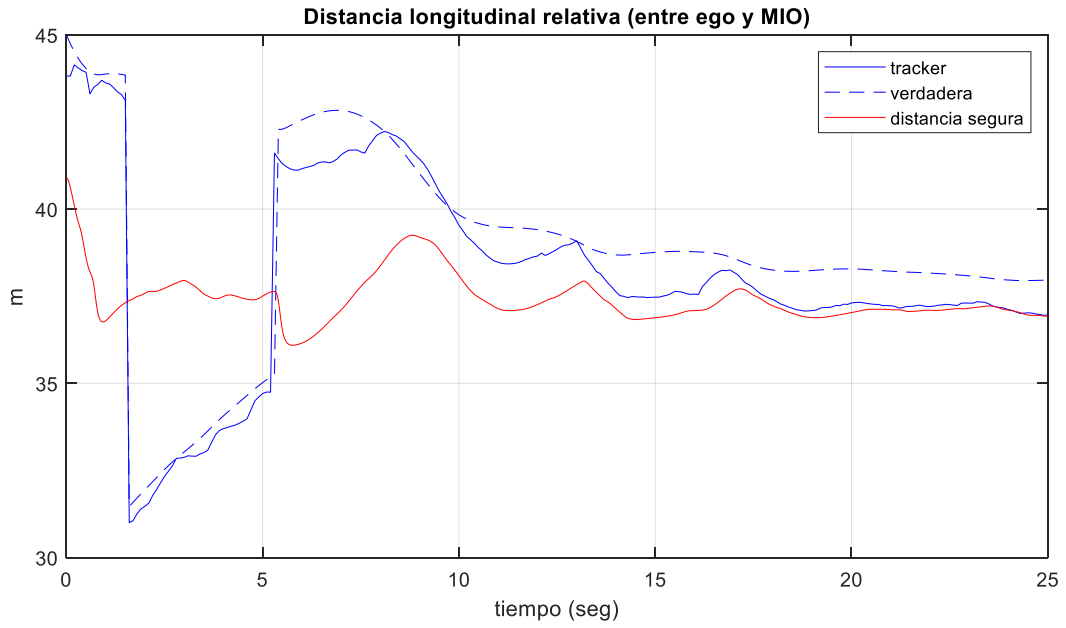


Figura 8.2.5.1. Escenario 05, comparación de los cuatro controladores y su efecto de la variación de la velocidad longitudinal del vehículo EGO.

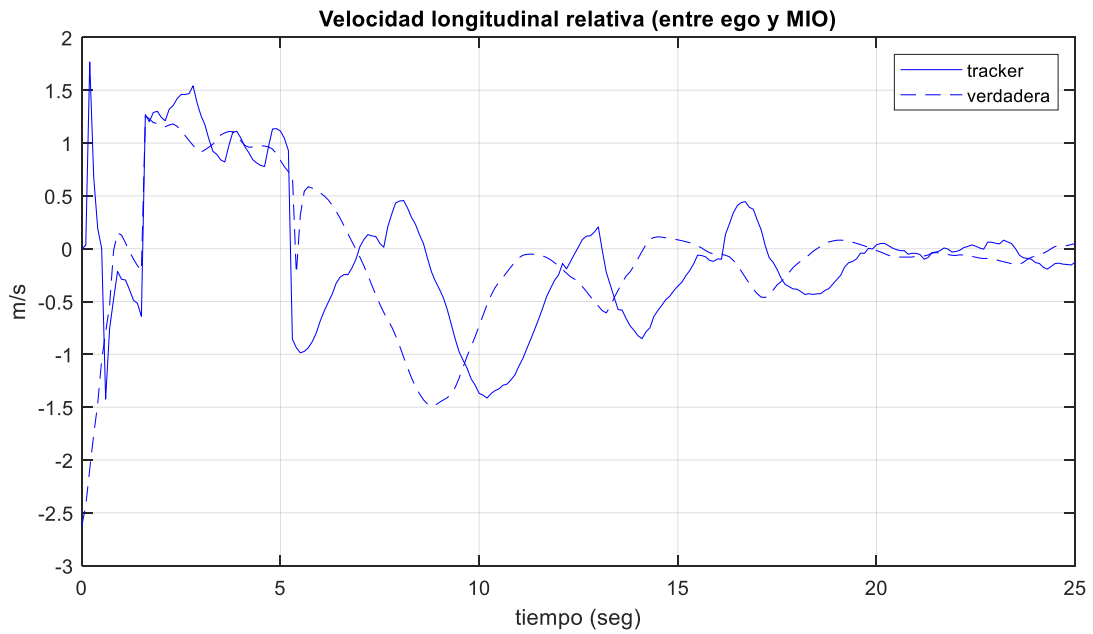


Si verificamos el comportamiento de cada controlador por separado se obtiene lo siguiente:

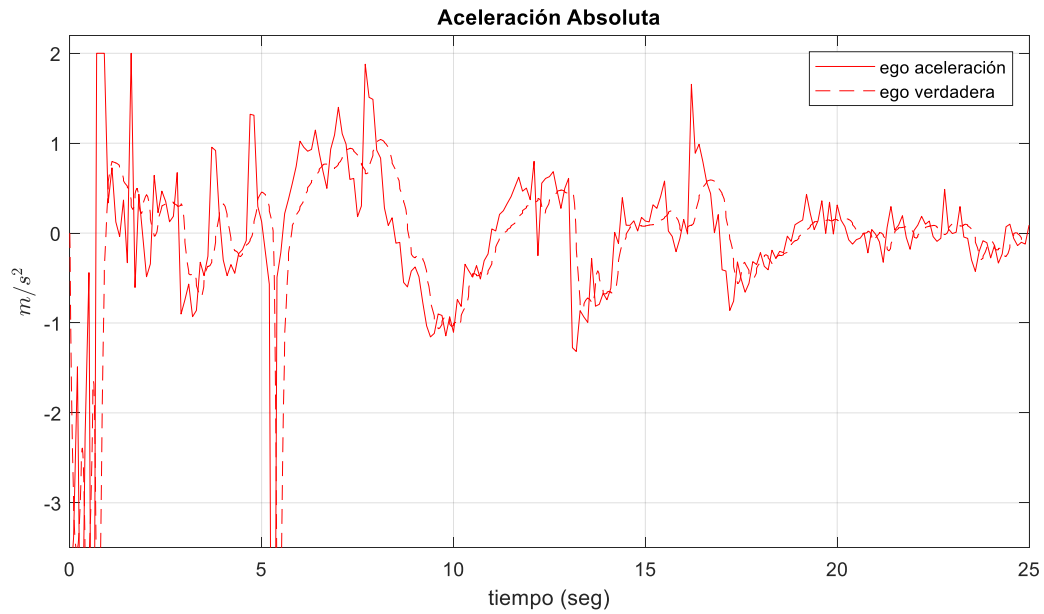
a)



b)



c)



d)

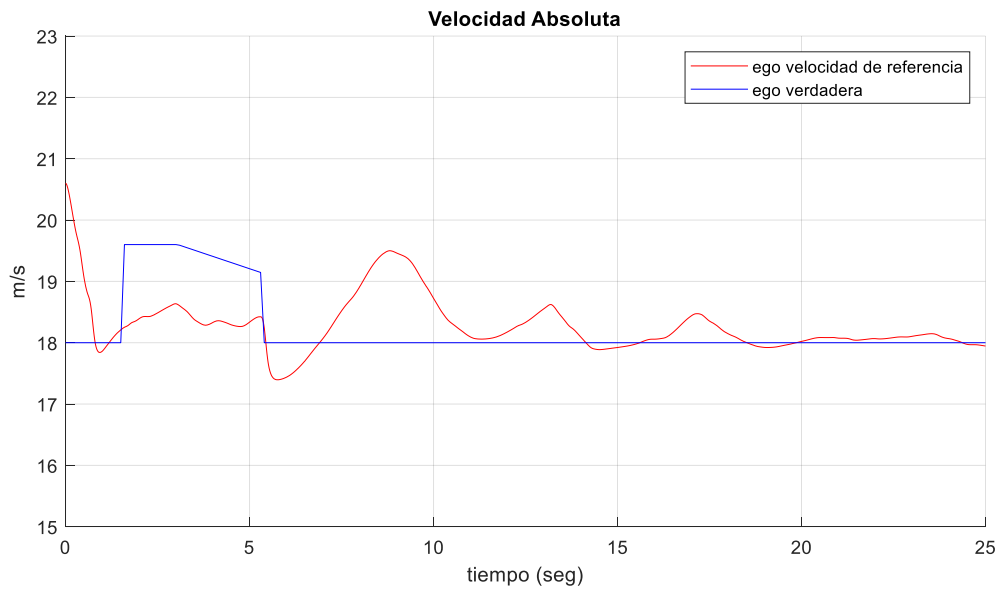
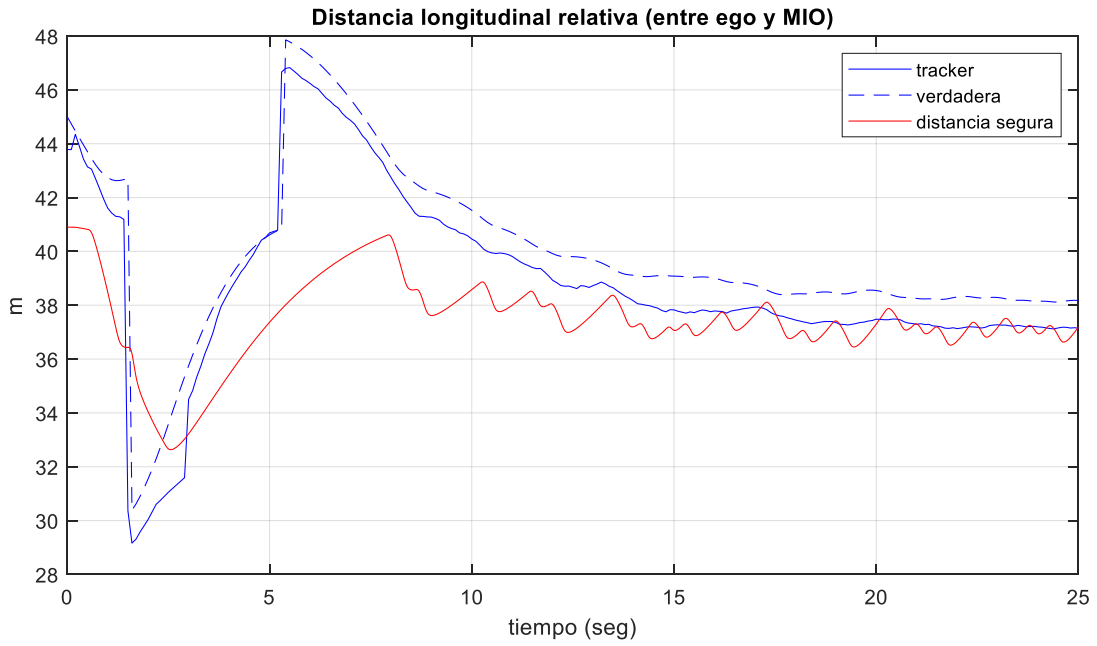


Figura 8.2.5.2. Desempeño del controlador PID de ACC, a) Distancia longitudinal relativa, b) velocidad longitudinal relativa, c) Aceleración absoluta, d) Velocidad absoluta

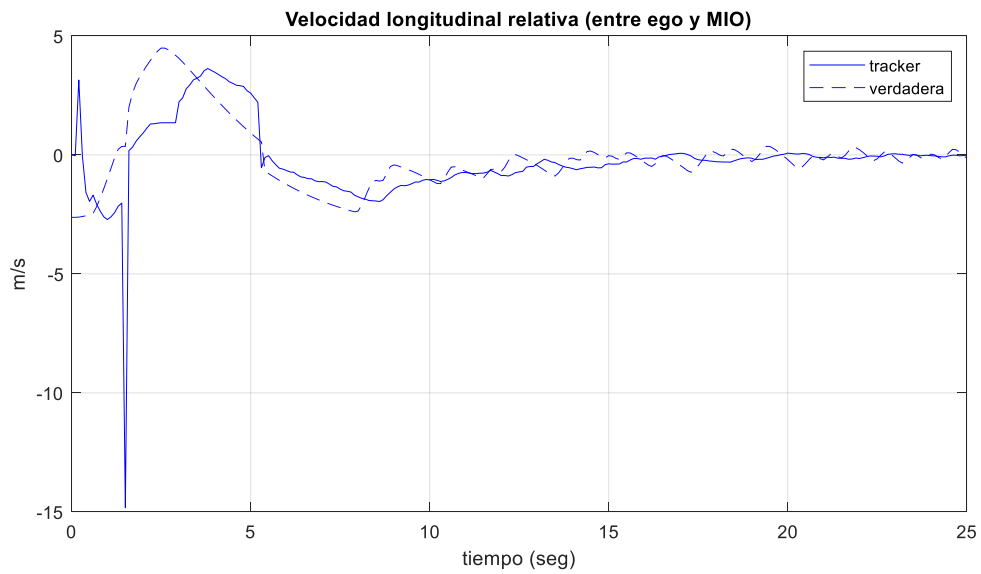
De la Figura 8.2.5.2, el controlador PID mantiene la distancia segura durante casi toda la simulación; además, su velocidad presenta un comportamiento bueno al reducir la velocidad y

ajustarse lentamente a la referencia, este ajuste es en un lapso de tiempo adecuado ya que logra reducirse en el tiempo de simulación.

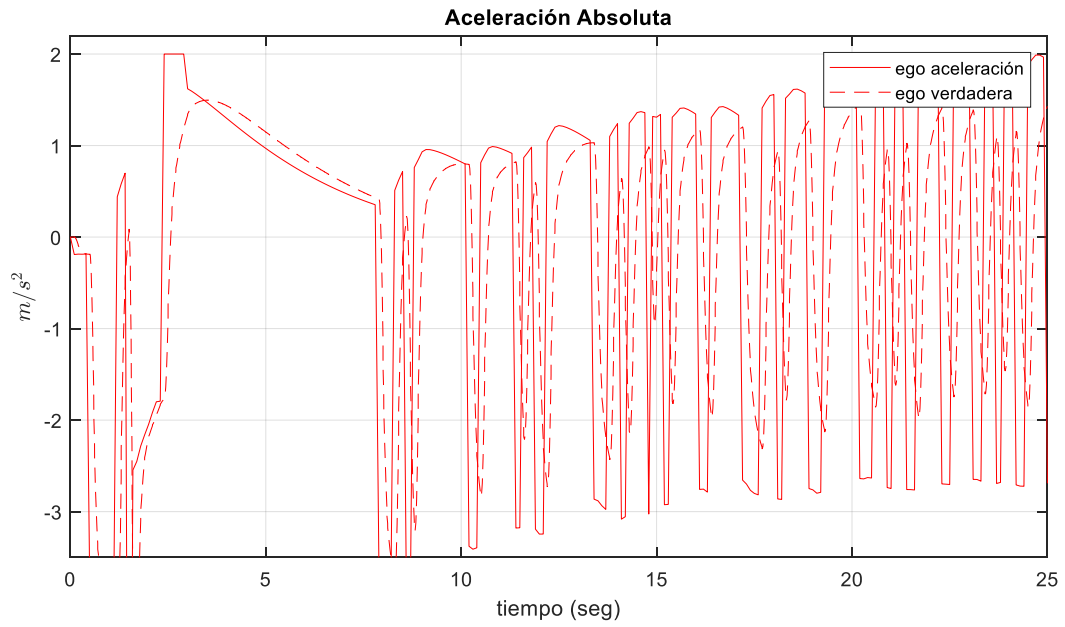
a)



b)



c)



d)

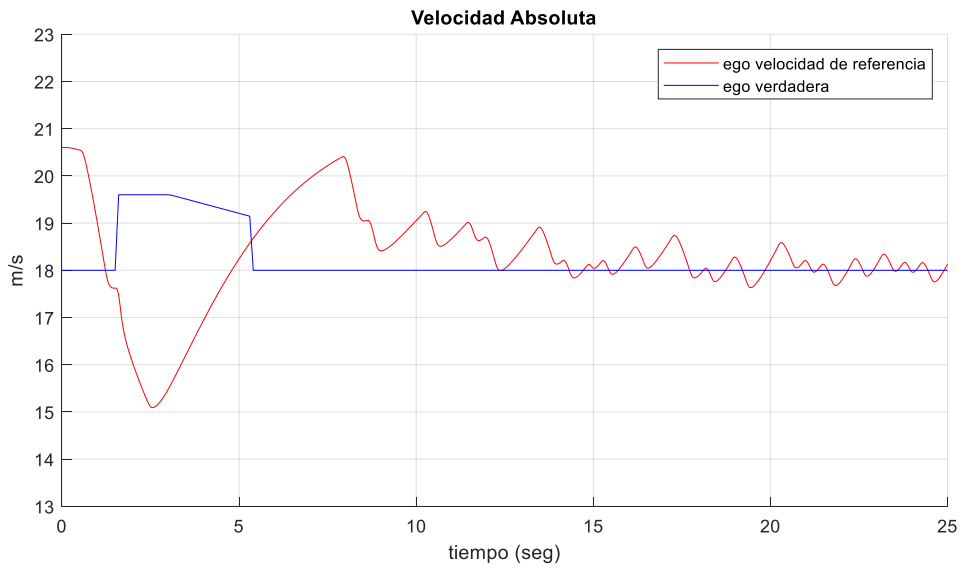
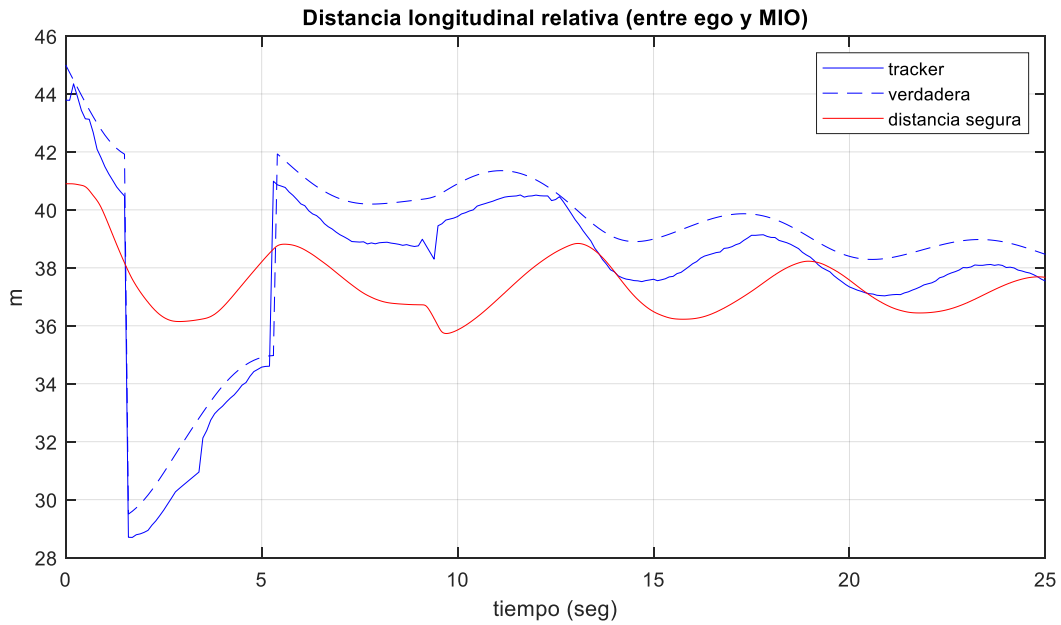


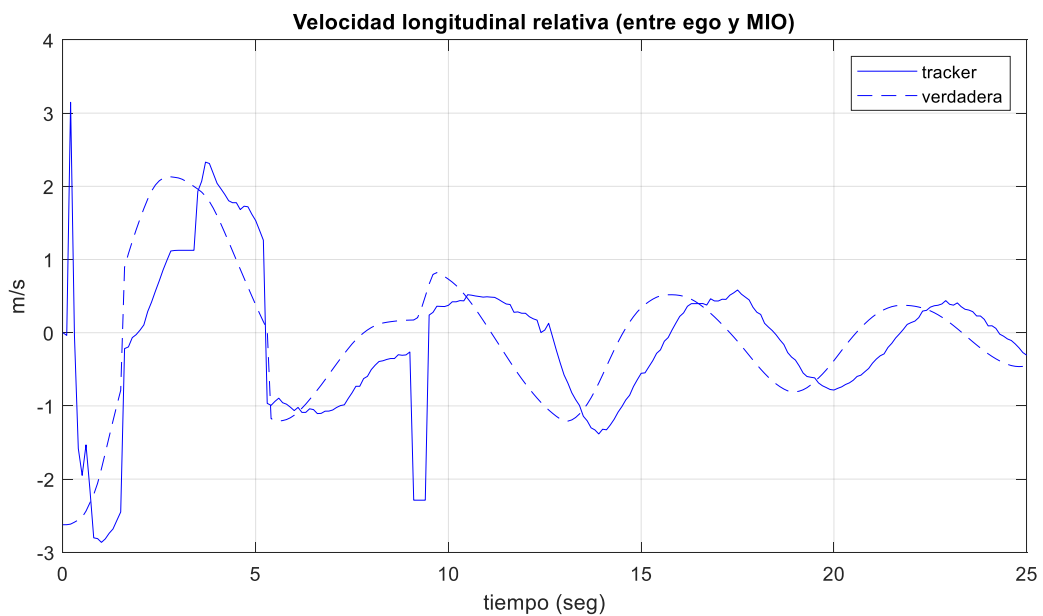
Figura 8.2.5.3. Desempeño del controlador SMC de ACC, a) Distancia longitudinal relativa, b) velocidad longitudinal relativa, c) Aceleración absoluta, d) Velocidad absoluta

De la Figura 8.2.5.3, el controlador SMC presenta un comportamiento adecuado ya que registra una distancia aceptable y un buen ajuste de velocidad en la mayoría de la simulación; sin embargo, presenta el comportamiento característico de “chattering” en la señal de aceleración.

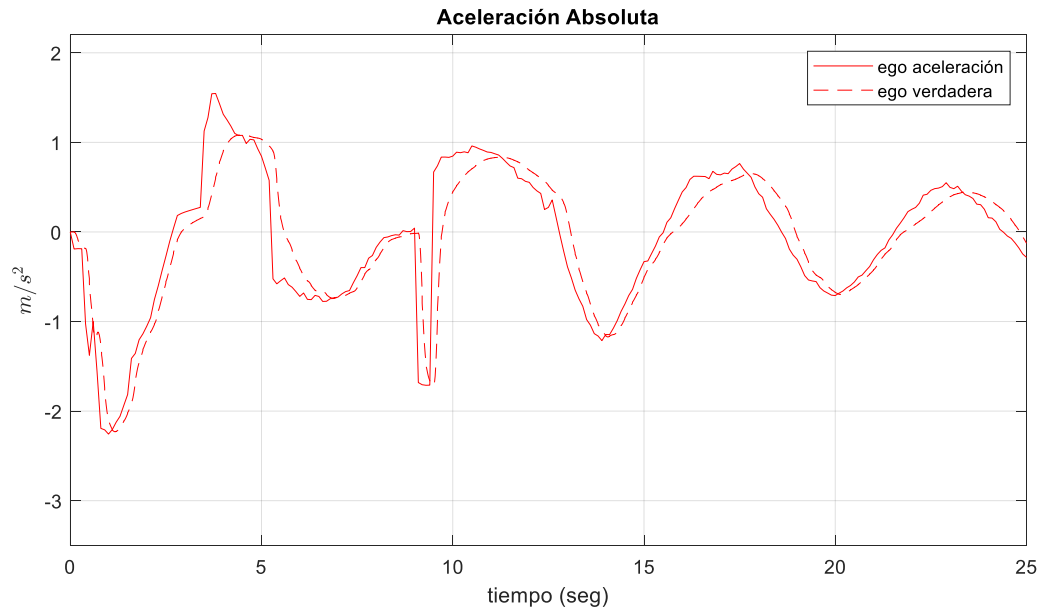
a)



b)



c)



d)

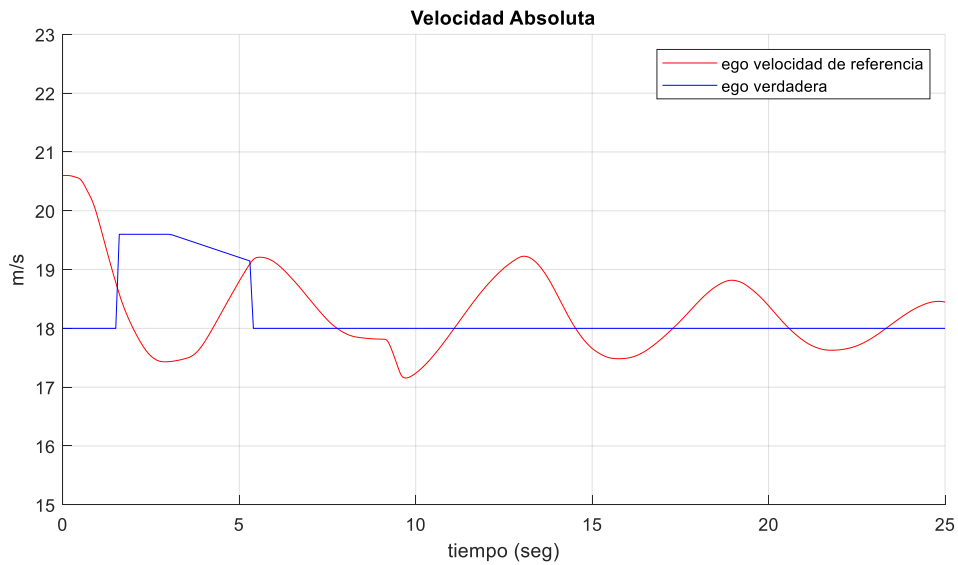
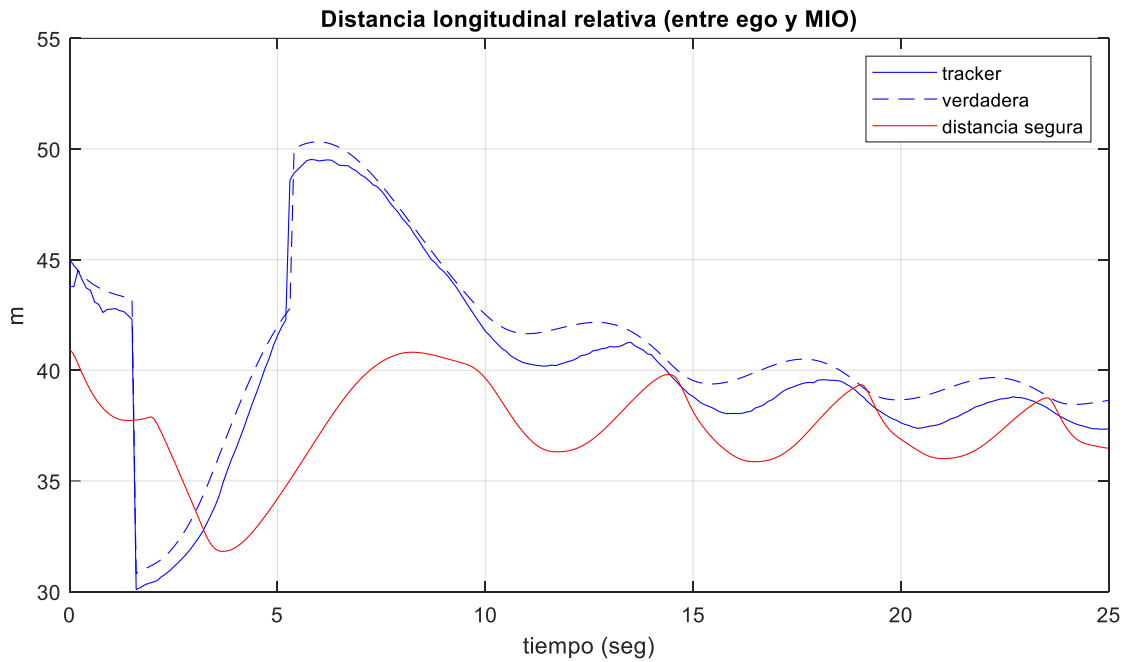


Figura 8.2.5.4. Desempeño del controlador CTG de ACC, a) Distancia longitudinal relativa, b) velocidad longitudinal relativa, c) Aceleración absoluta, d) Velocidad absoluta

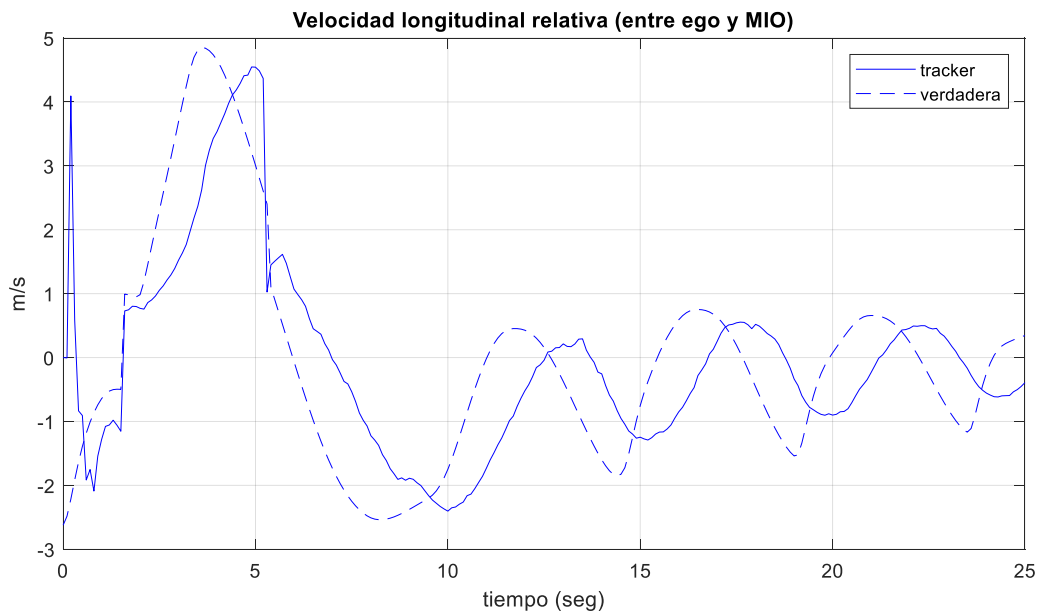
De la Figura 8.2.5.4, el controlador CTG parece ser que en el instante de la perturbación frena con cierta magnitud y posteriormente intenta ajustarse a la velocidad deseada, pero la oscilación se mantiene con una amplitud grande hasta casi finalizar la simulación, en donde esta

comienza a presentar un comportamiento decreciente. En cuanto a la distancia, aunque oscilante, se mantiene por encima del límite requerido con excepción del primer instante de frenado, esto puede ser un indicio de respuesta tardía por parte del controlador.

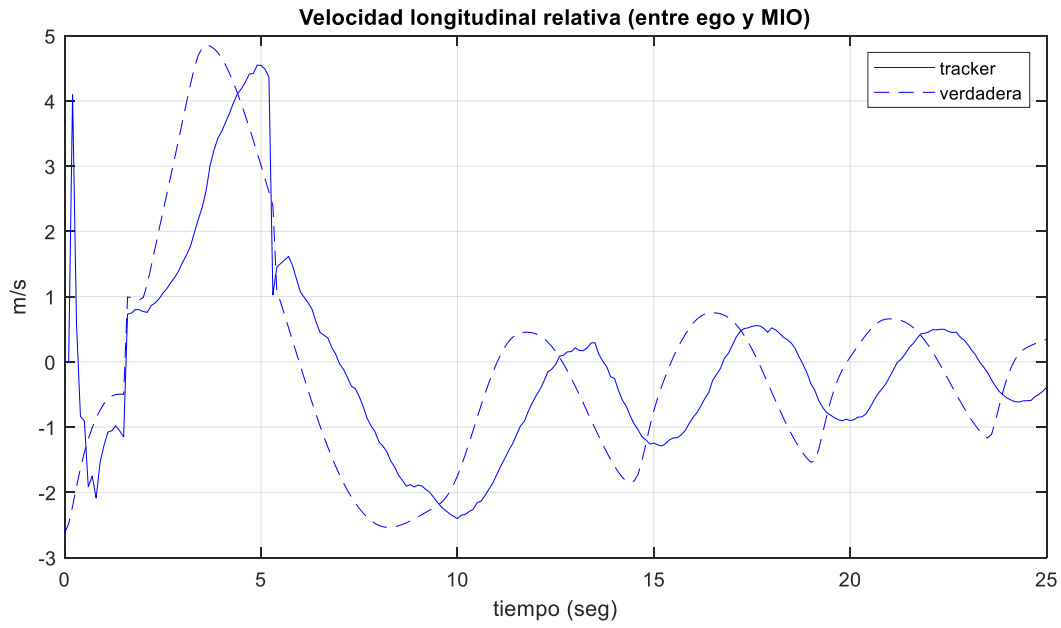
a)



b)



c)



d)

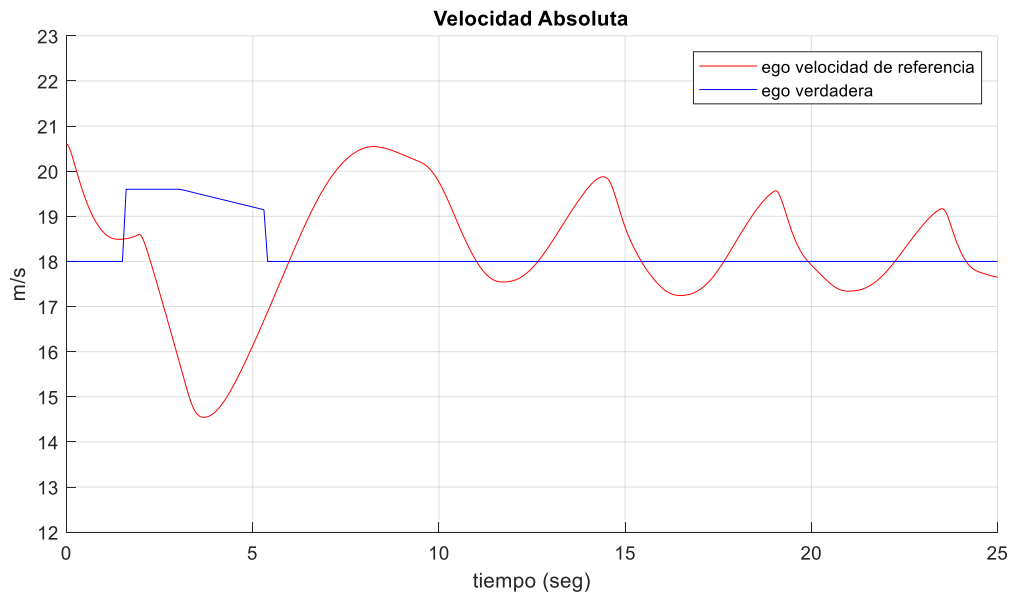
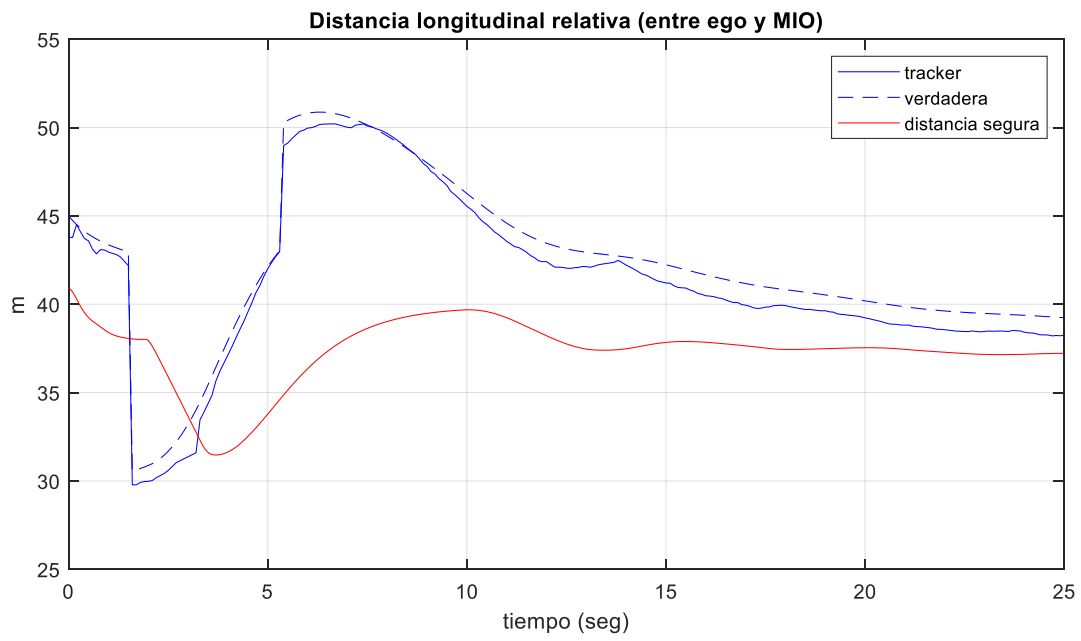


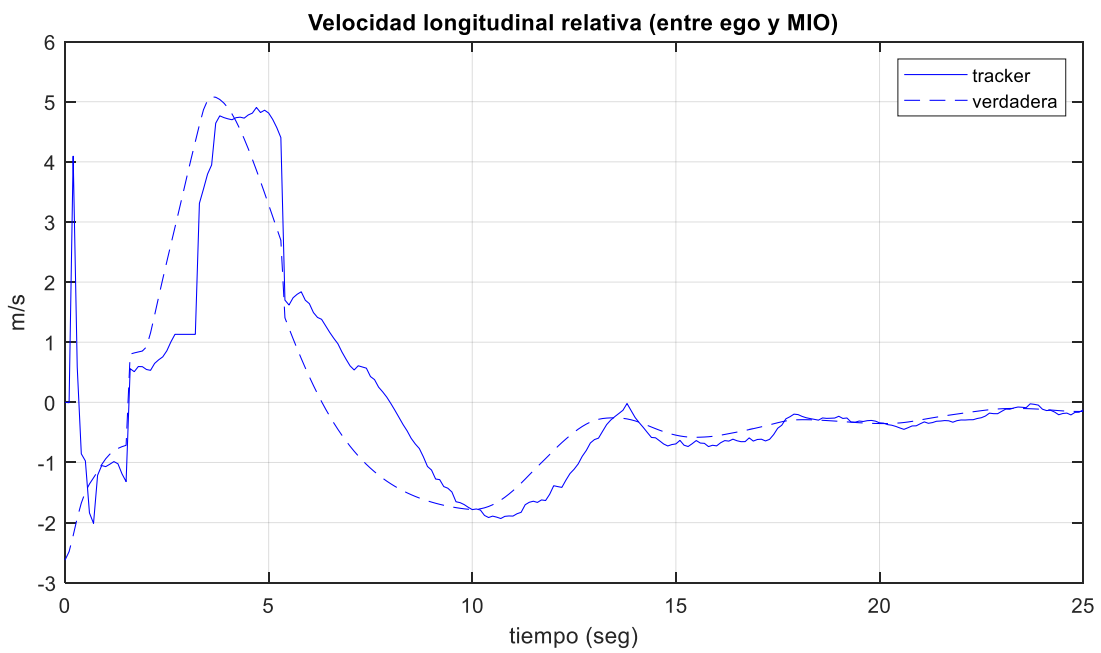
Figura 8.2.5.5. Desempeño del controlador MPC con  $H_p$  de 20 del ACC, a) Distancia longitudinal relativa, b) velocidad longitudinal relativa, c) Aceleración absoluta, d) Velocidad absoluta



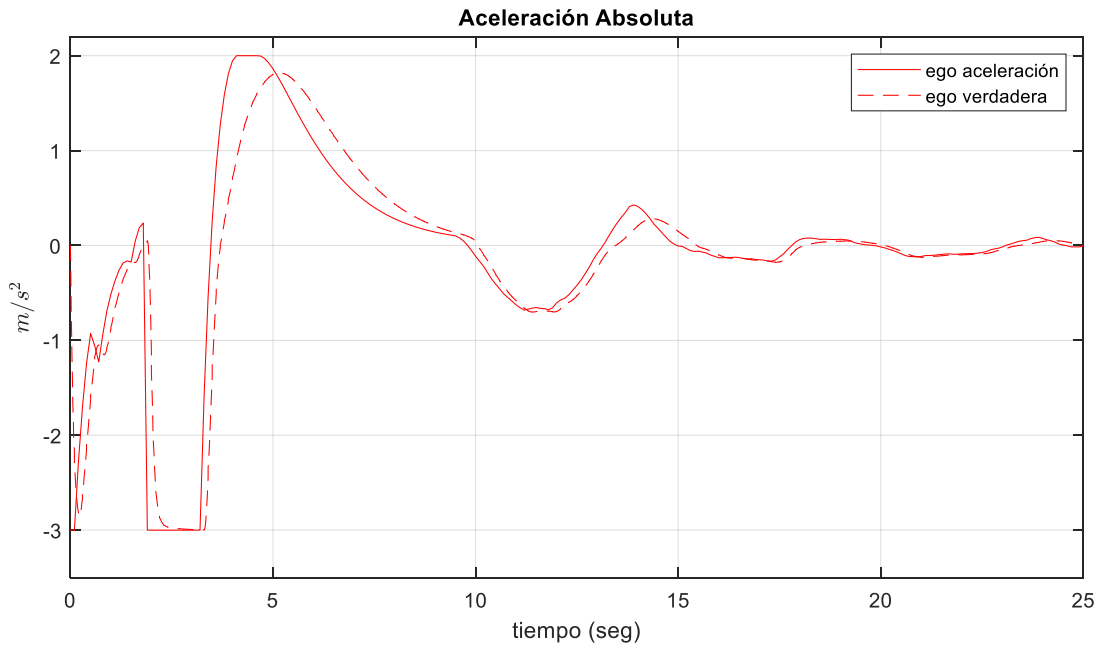
a)



b)



c)



d)

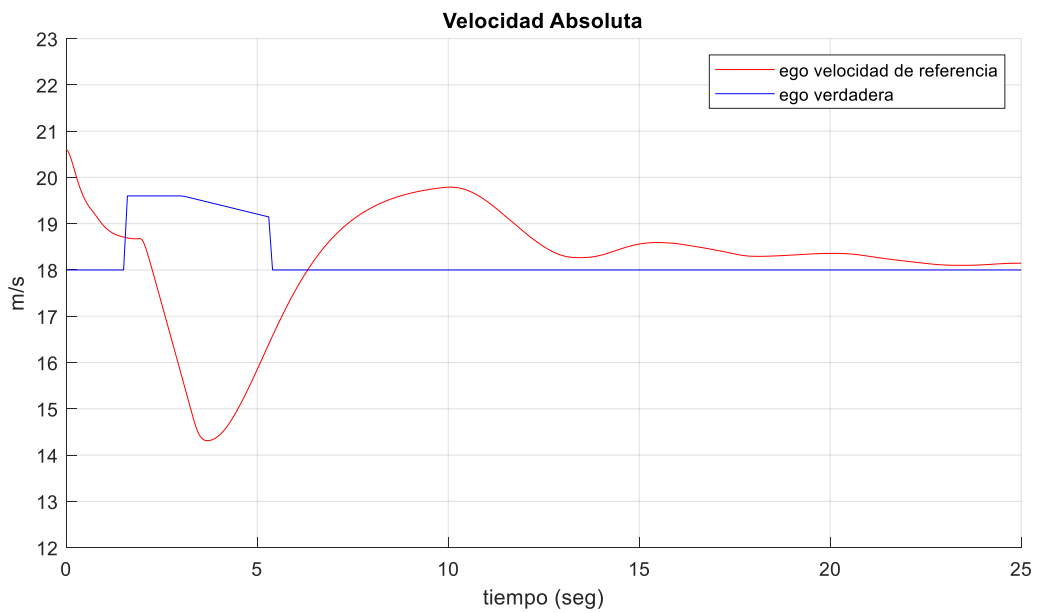


Figura 8.2.5.6. Desempeño del controlador MPC con  $H_p$  de 40 del ACC, a) Distancia longitudinal relativa, b) velocidad longitudinal relativa, c) Aceleración absoluta, d) Velocidad absoluta

Ahora bien, de la Figura 8.2.5.5 y 8.2.5.6 se observa el comportamiento del controlador MPC con Hp de 20 y 40 respectivamente. En el primer caso, observamos que el comportamiento es muy similar al descrito en el CTG, con la excepción de que decrece la amplitud un poco más rápido. Por otro lado, para el Hp de 40, se puede observar un comportamiento adecuado a la situación descrita en este escenario, ya que presenta prácticamente una distancia segura durante toda la simulación y se aproxima adecuadamente a la velocidad de referencia manteniendo una buena variación en la aceleración del vehículo.

Para continuar con el análisis de este escenario se realizó el cálculo del error RMSE, mostrado en la Figura 8.2.5.7; así como, las Tablas 8.2.5.1 y 8.2.5.2 que nos ayudan a evaluar el desempeño de los controladores.

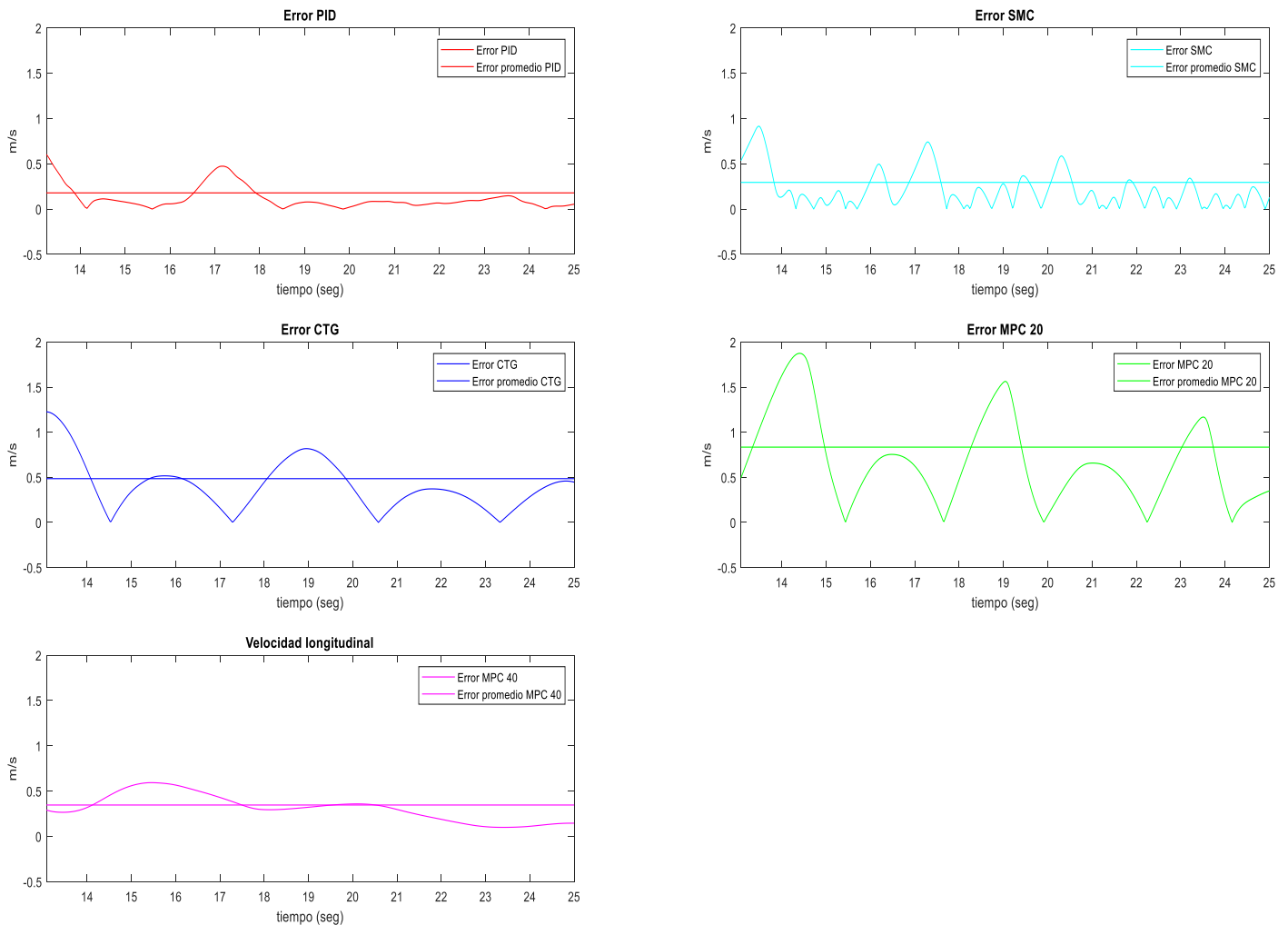


Figura 8.2.5.7. Variación del error RMSE de cada controlador con respecto a la velocidad de referencia

ESCENARIO 5									
Controlador	Vmax	%V	t1	$\tau$	te	RMSE	%RMSE vel	Tiempo de frenado	Velocidad en el tiempo de frenado
Clásico PID	19.48	8.22	6.95	6.60	18.04	0.179	0.99	0.9	17.8465
SMC	20.4	13.33	4.76	4.00	17.72	0.2945	2.27	2.52	15.0977
CTG	19.2	6.67	4.26	4.00	20.64	0.4852	3.73	2.94	17.43
MPC 20	20.54	14.11	5.94	5.34	19.9	0.8349	6.42	3.70	14.54
MPC 40	19.78	9.89	6.26	5.44	20.04	0.3664	2.82	3.64	14.31

Tabla 8.2.5.1. Comparación del rendimiento de los controladores diseñados, escenario 04.

De la Tabla 8.2.5.1 se observa que la mayoría de controladores tienen un error relativamente alto con excepción del PID, SMC y el MPC con Hp de 40. También, los cuatro controladores presentan un valor alto de sobre pico después de frenar súbitamente cuando el vehículo adyacente cambia de carril, mientras esto sucede, la principal diferencia recae en el tiempo de respuesta ante el frenado y el ajuste de velocidad donde el SMC y CTG tiene los tiempos de reacción más rápido. En general se puede observar el índice de ponderación de la Tabla 8.2.5.2.

EVALUACIÓN DEL CONTROLADOR									
Controlador	Vmax	%V	t1	$\tau$	te	RMSE	%RMSE vel	Puntuación total	Orden de desempeño
<b>Ponderación (0-1)</b>	1	1	3	3	5	5	5		
Clásico PID	19.48	8.22	20.85	19.80	90.20	0.89	4.96	164.41	2
SMC	20.40	13.33	14.28	12.00	88.60	1.47	11.33	161.41	1
CTG	19.20	6.67	12.78	12.00	103.20	2.43	18.66	174.93	3
MPC 20	20.54	14.11	17.82	16.02	99.50	4.17	32.11	204.28	5
MPC 40	19.78	9.89	18.78	16.32	100.20	1.83	14.09	180.89	4

Tabla 8.2.5.2. Índice de ponderación para seleccionar el controlador con el mejor desempeño, escenario 05.

De la Tabla anterior se concluye que el controlador SMC tiene el mejor desempeño, seguido del PID y el CTG. Esto se debe a que es el controlador que tuvo un mejor ajuste en de la distancia y velocidad en el menor periodo de tiempo. Con respecto a los dos controladores restantes la diferencia resulta ser muy pequeña; sin embargo, este acomodo se debe tanto al porcentaje de sobre pico, como al tiempo que tarda en alcanzar la señal deseada.

### 8.2.6 Escenario 6

Tal y como se describió en la sección de escenarios, este en particular se compone del vehículo EGO frenando en primera instancia y posteriormente alcanzando la velocidad de referencia según sea el caso, el ajuste de velocidad realizado por los cuatro controladores se puede observar en la Figura 8.2.6.1.

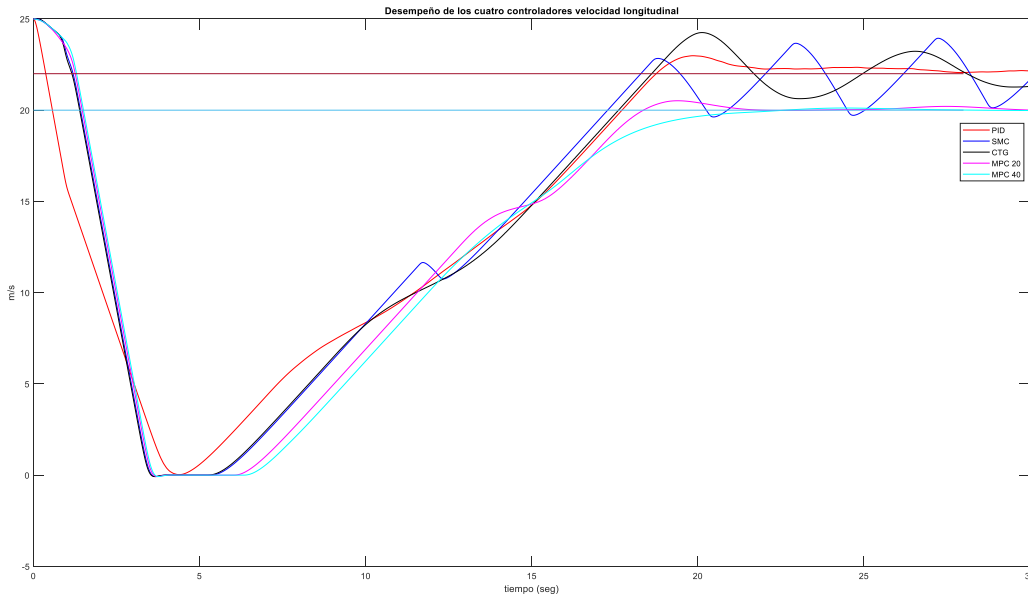
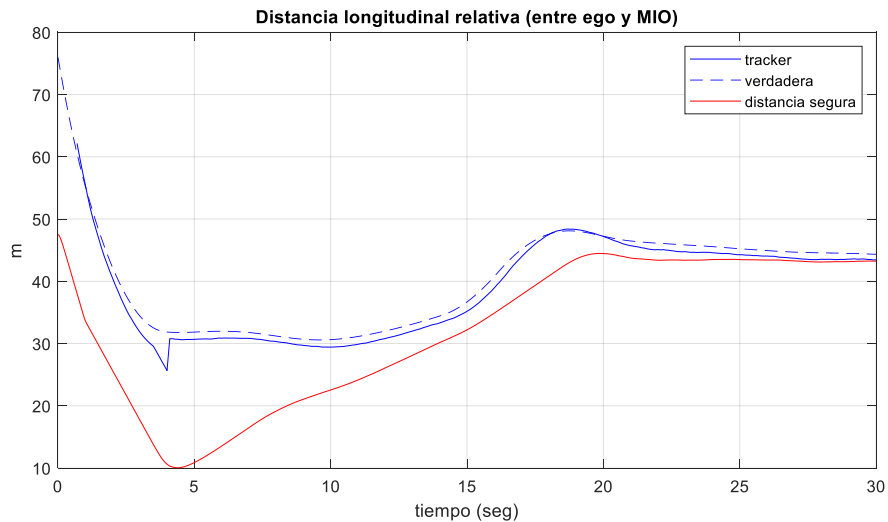


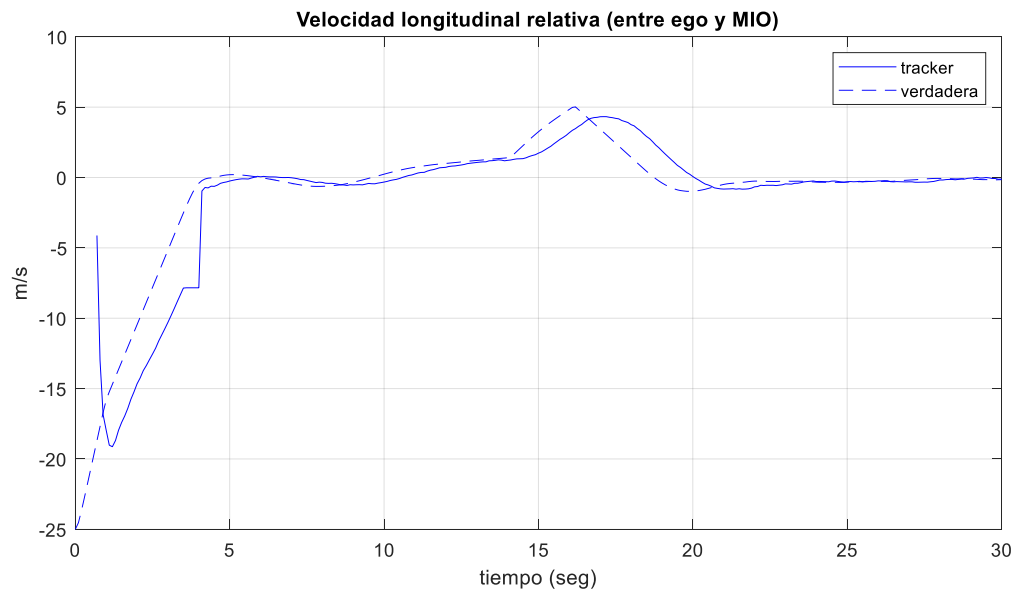
Figura 8.2.6.1. Escenario 06, comparación de los cuatro controladores y su efecto de la variación de la velocidad longitudinal del vehículo EGO.

Verificando el comportamiento individual de cada controlador se obtiene lo siguiente:

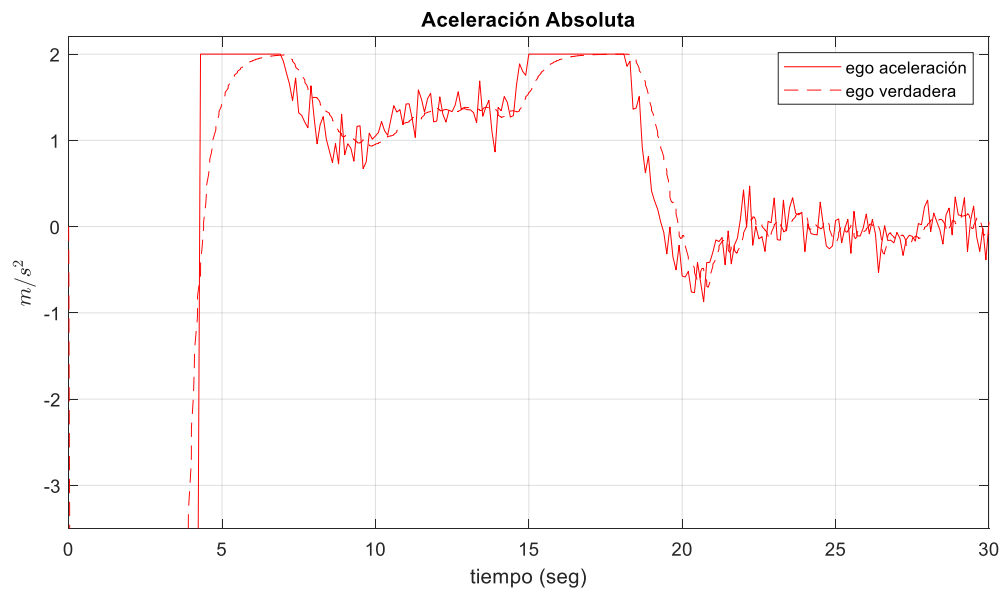
a)



b)



c)



d)

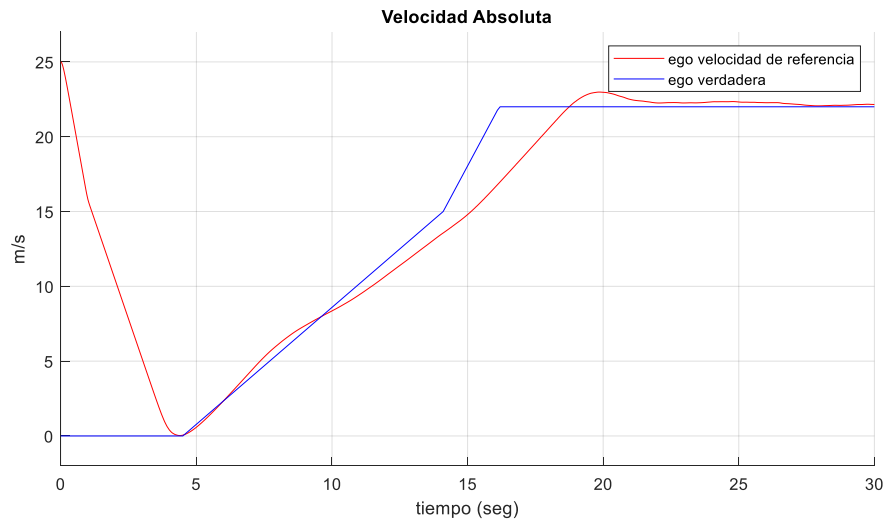
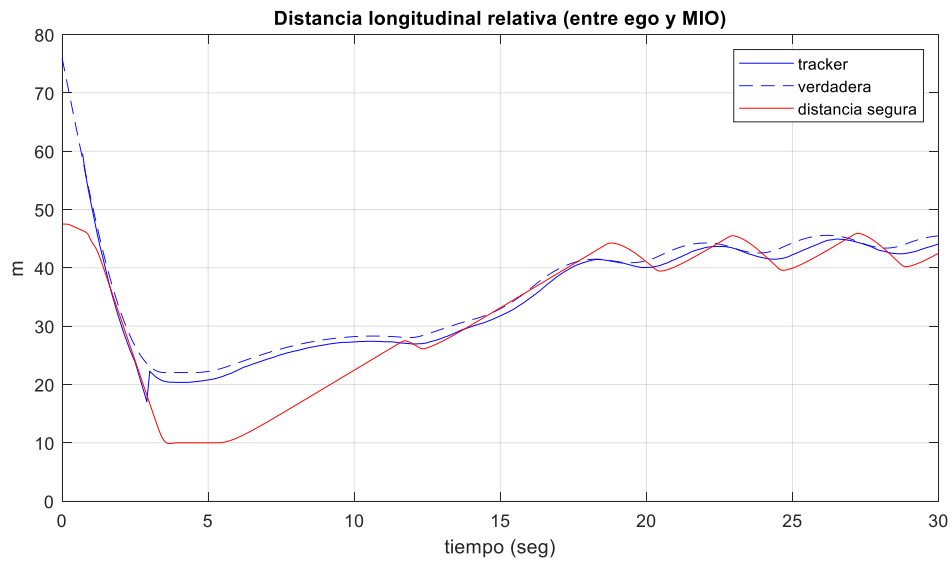


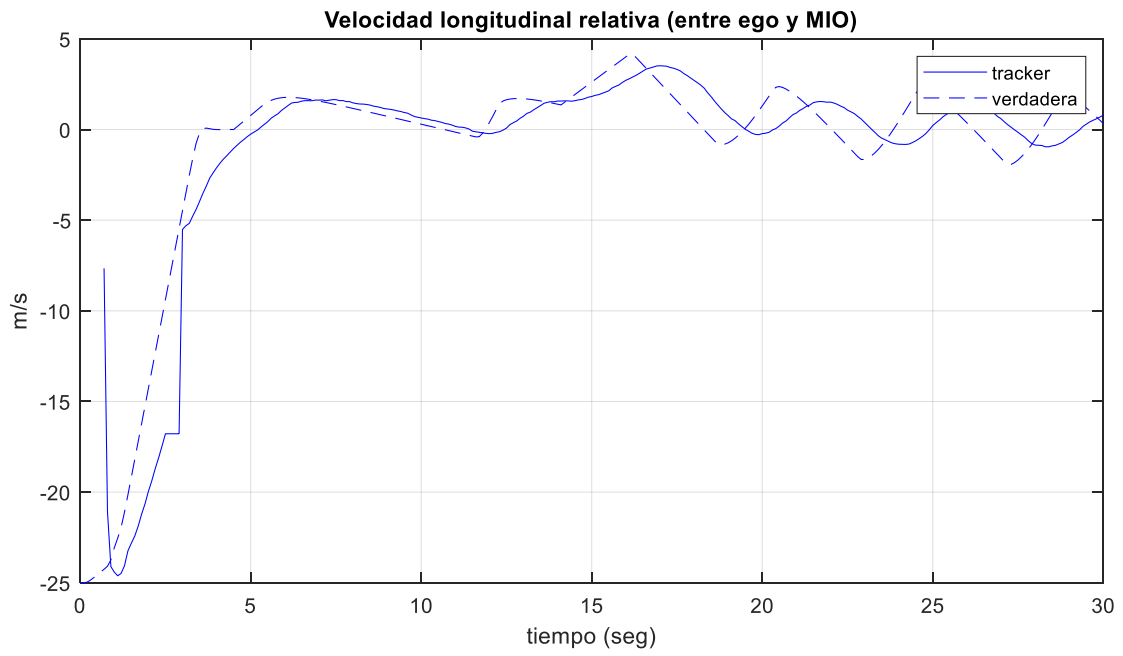
Figura 8.2.6.2. Desempeño del controlador PID de ACC, a) Distancia longitudinal relativa, b) velocidad longitudinal relativa, c) Aceleración absoluta, d) Velocidad absoluta

De la Figura 8.2.6.2, el controlador PID muestra una buena respuesta durante el frenado. Posteriormente podemos observar, en cuanto incrementa la velocidad, presenta un sobre pico mínimo en la velocidad, esto se debe a que la señal proporcionada de aceleración se mantiene prácticamente constante desde la mitad de la simulación, no obstante, se observa que con el tiempo se comienza a regular el valor de referencia, con valores de amplitud bastante pequeños.

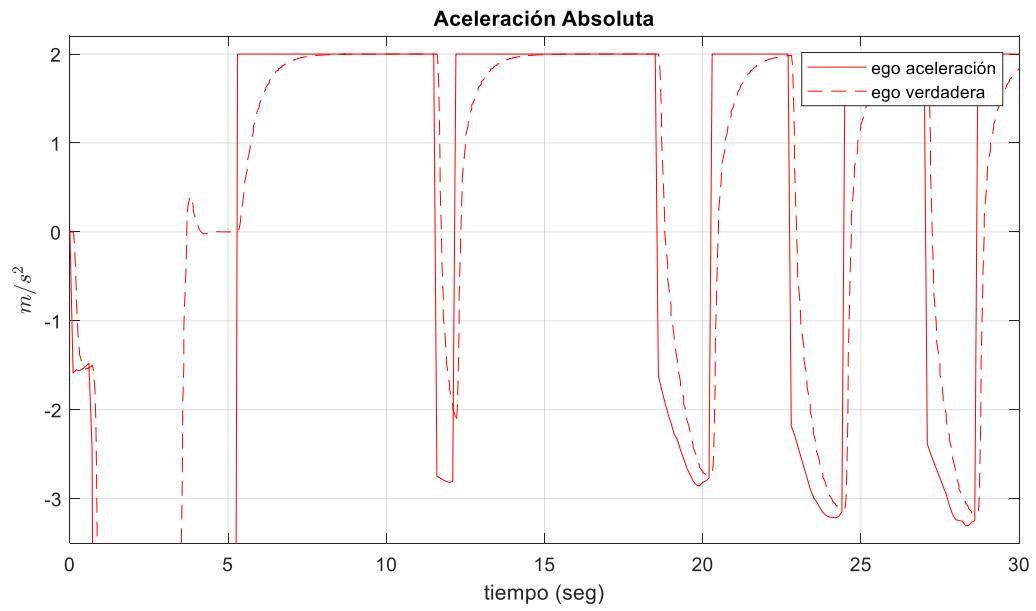
a)



b)



c)





d)

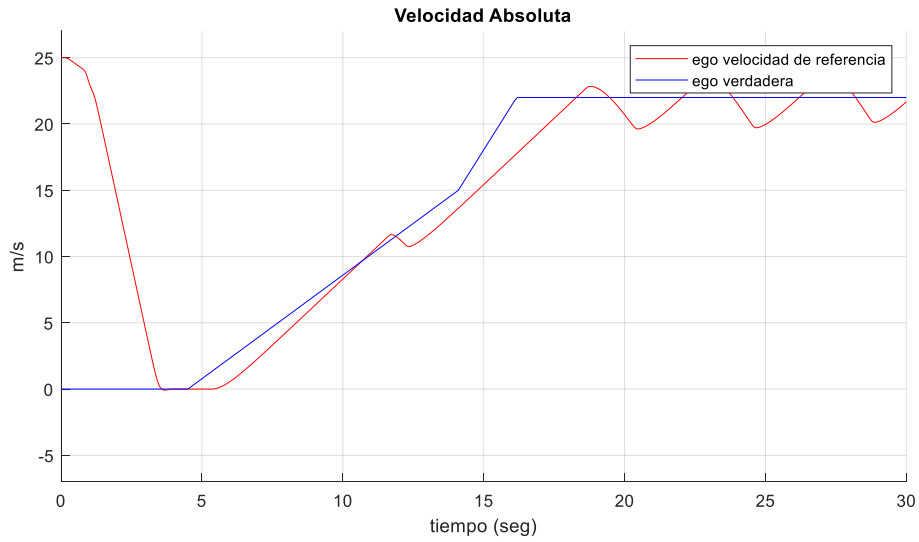
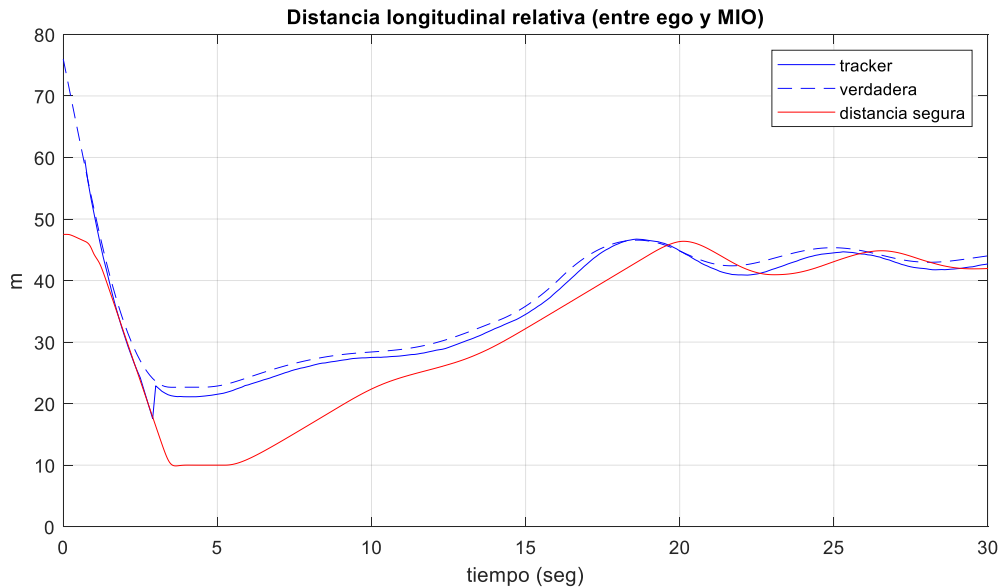


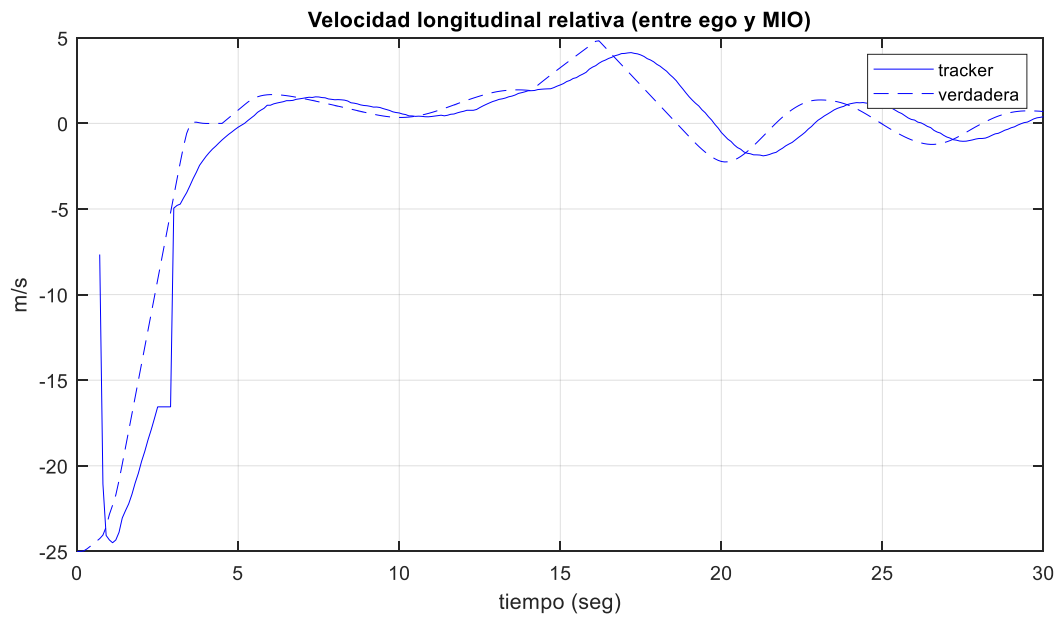
Figura 8.2.6.3. Desempeño del controlador SMC de ACC, a) Distancia longitudinal relativa, b) velocidad longitudinal relativa, c) Aceleración absoluta, d) Velocidad absoluta

De la Figura 8.2.6.3, el controlador SMC mantiene una distancia segura durante la mayor parte de la simulación, de igual forma presenta el comportamiento de “chattering” en menor medida que los escenarios anteriores y ajusta su velocidad a la referencia del vehículo MIO oscilando con una tendencia a reducir la amplitud para converger al valor deseado.

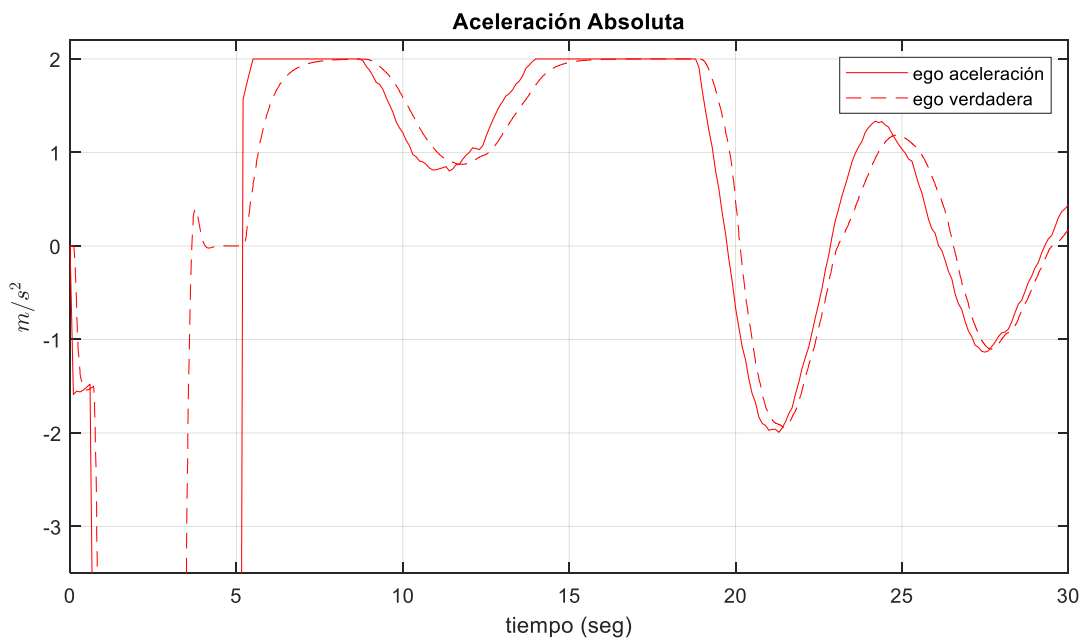
a)



b)



c)



d)

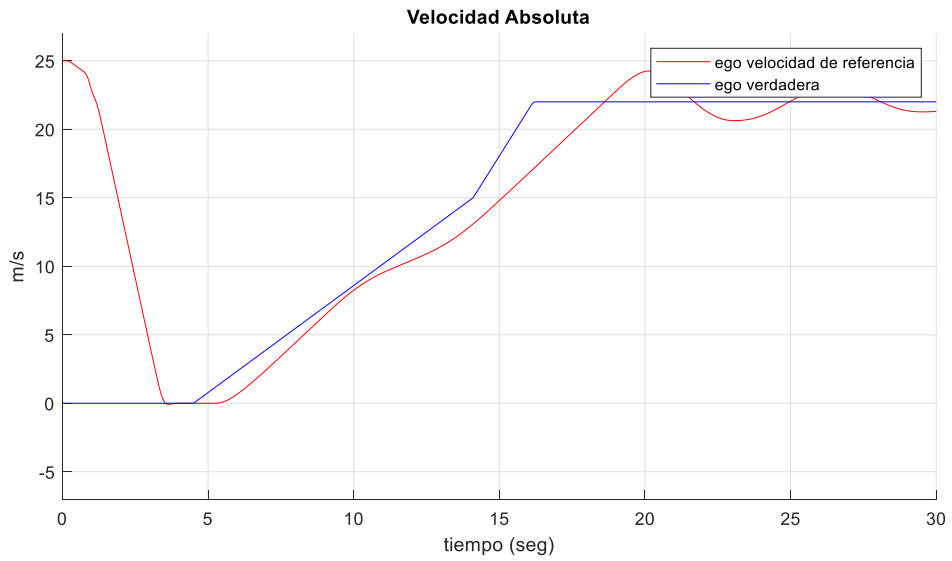
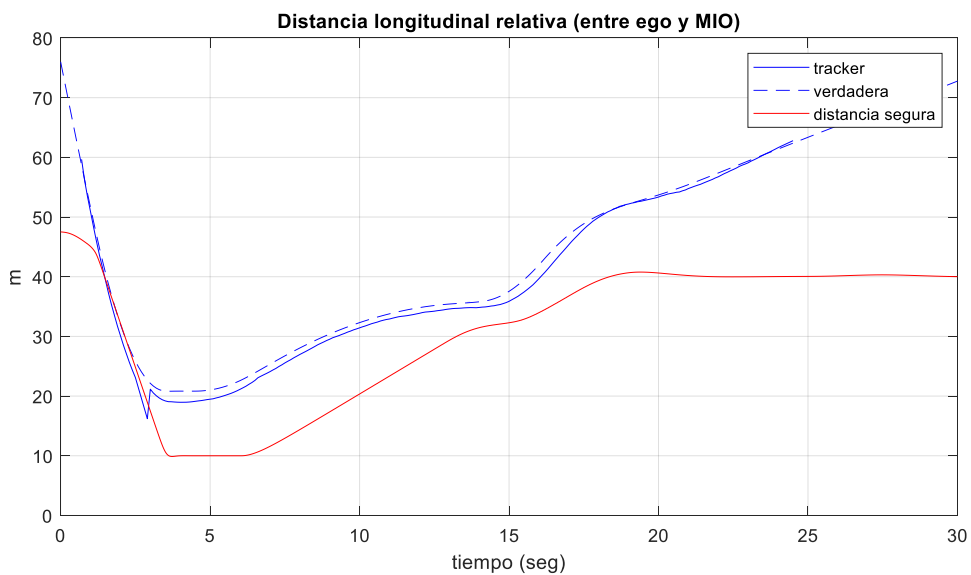


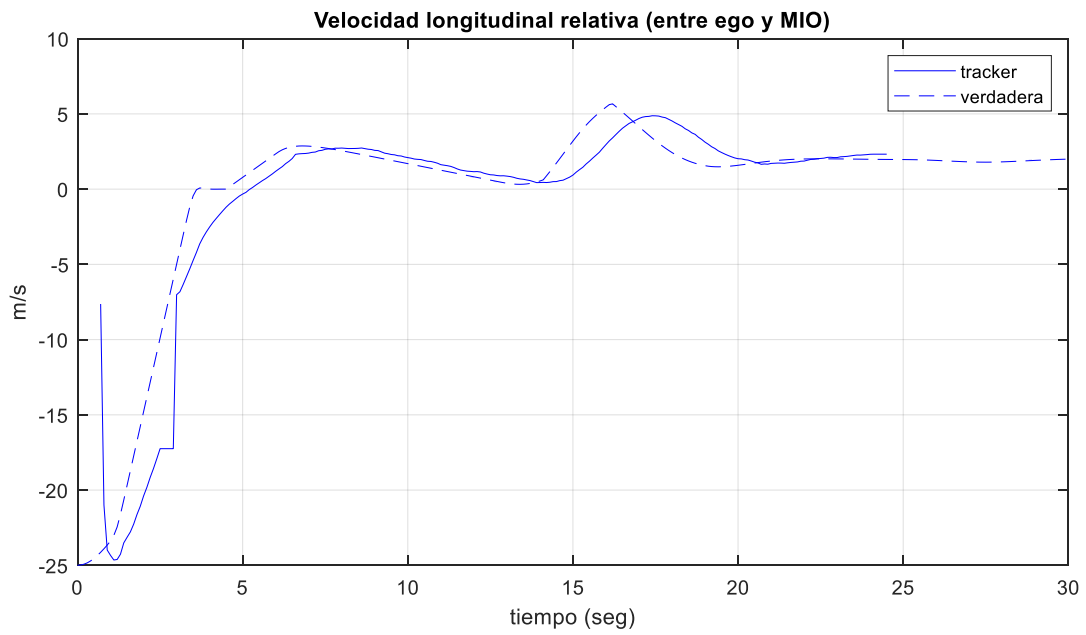
Figura 8.2.6.4. Desempeño del controlador CTG de ACC, a) Distancia longitudinal relativa, b) velocidad longitudinal relativa, c) Aceleración absoluta, d) Velocidad absoluta

De la Figura 8.2.6.4, el controlador CTG cumple con mantener una distancia segura con una aceleración paulatina hasta regular la velocidad con un sobrepaso aceptable. Además, podemos observar que en ningún momento la distancia calculada sobrepasa el rango del radar que está detectando el vehículo MIO, por lo cual la referencia de velocidad a la que se ajusta es la velocidad misma del vehículo MIO, caso contrario al PID y el MPC.

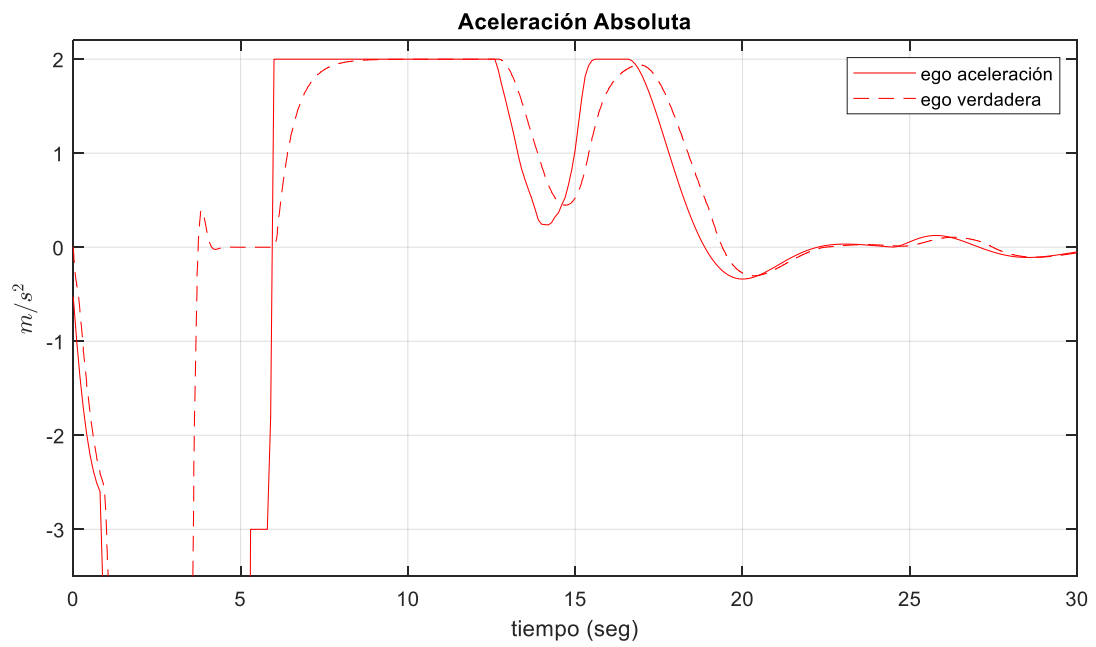
a)



b)



c)



d)

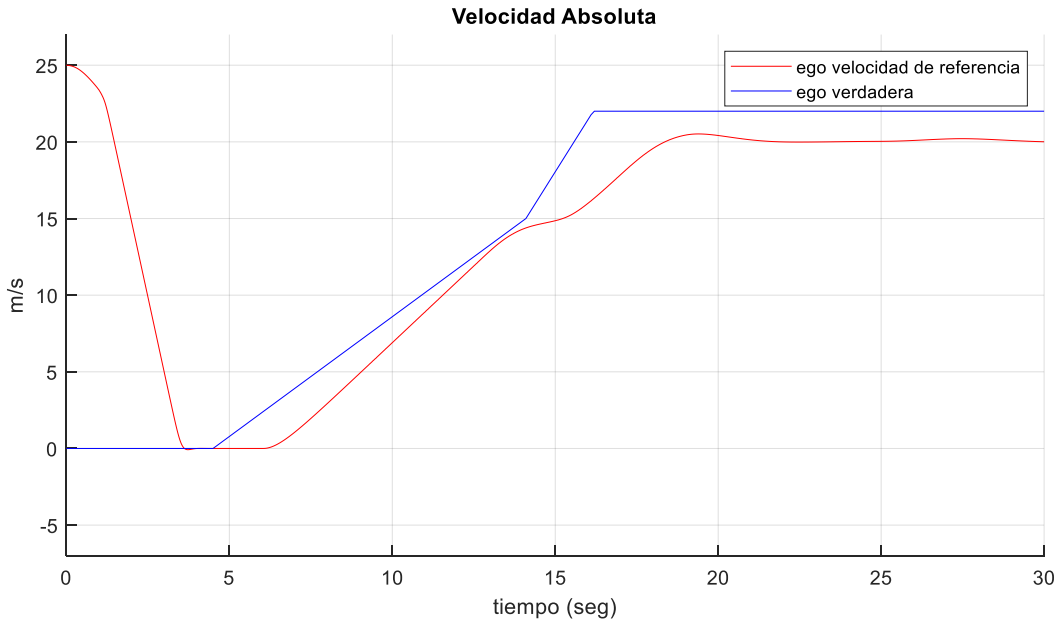
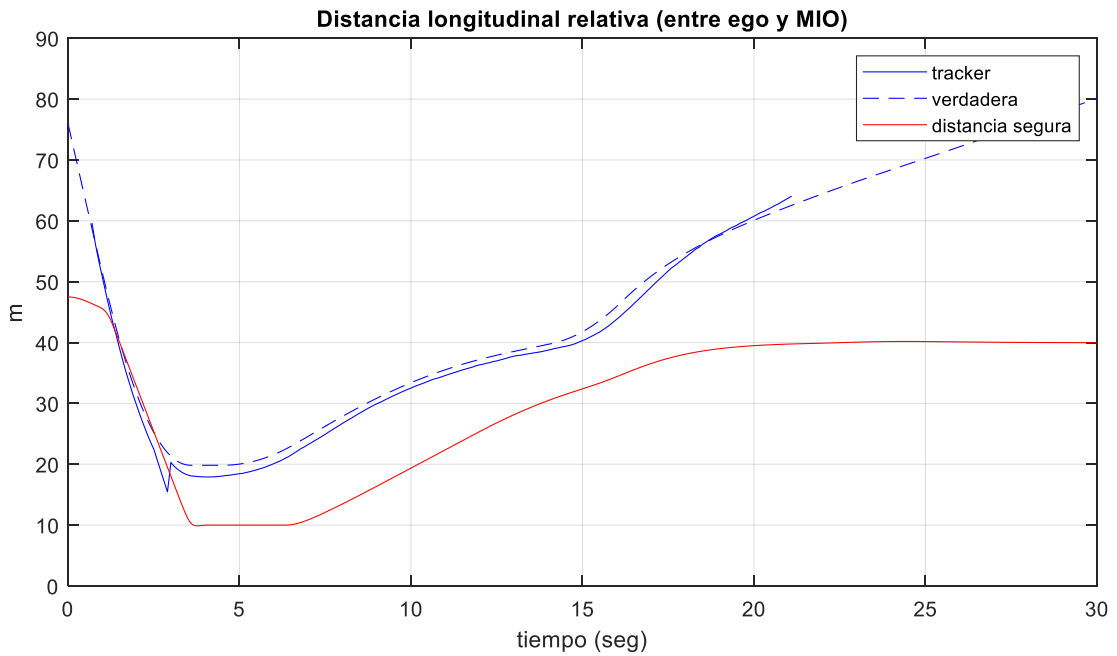
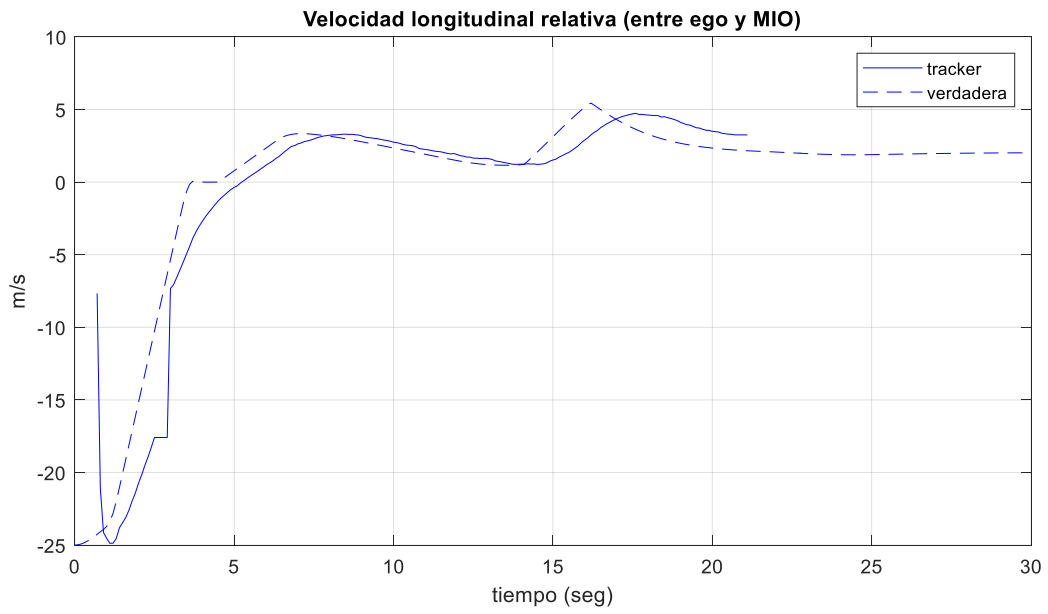


Figura 8.2.6.5. Desempeño del controlador PID de ACC, a) Distancia longitudinal relativa, b) velocidad longitudinal relativa, c) Aceleración absoluta, d) Velocidad absoluta

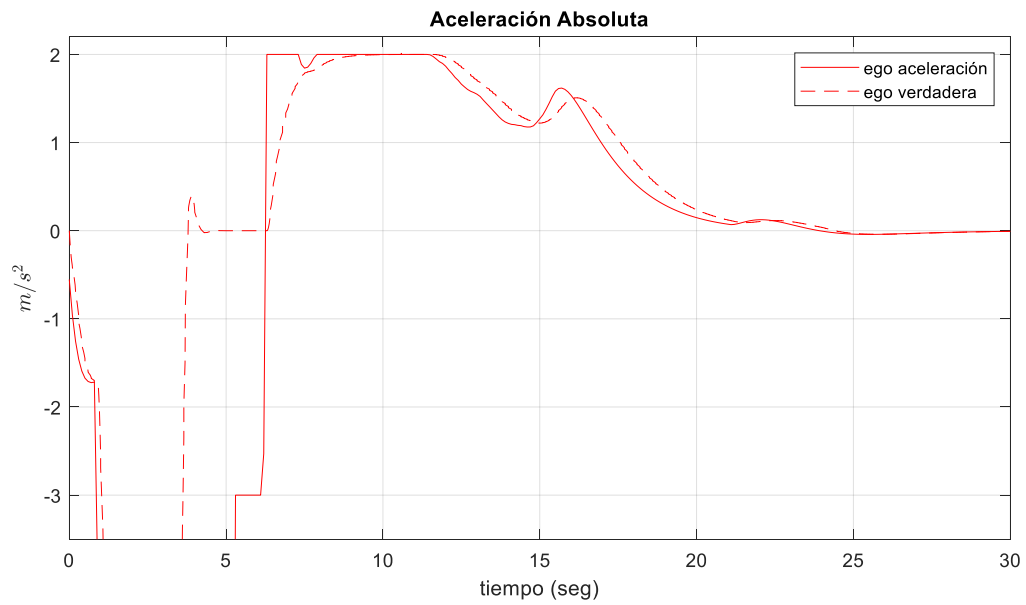
a)



b)



c)



d)

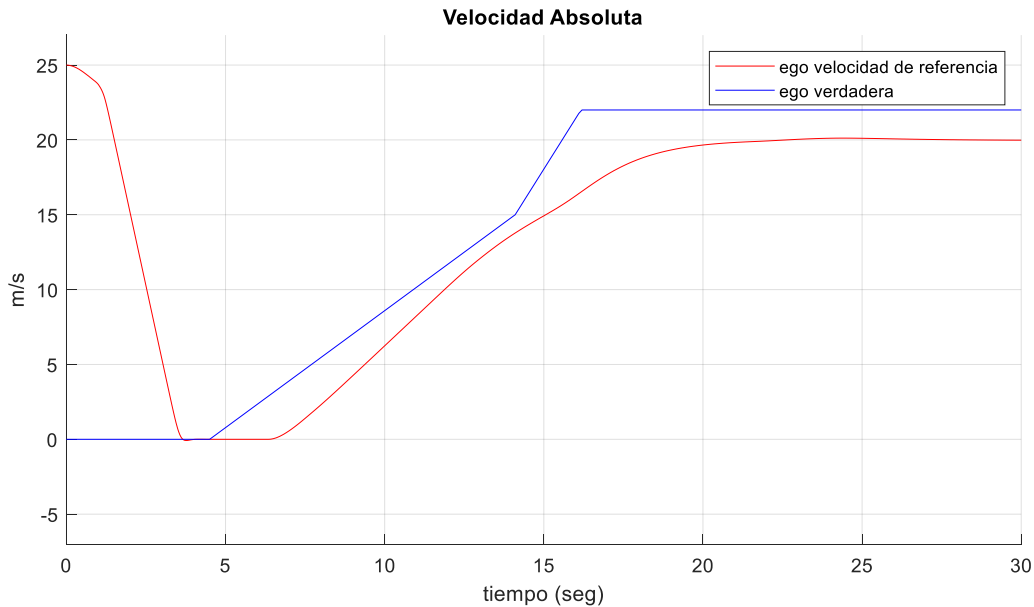


Figura 8.2.6.6. Desempeño del controlador PID de ACC, a) Distancia longitudinal relativa, b) velocidad longitudinal relativa, c) Aceleración absoluta, d) Velocidad absoluta

De las Figuras 8.2.6.5 y 8.2.6.6, el controlador MPC con  $H_p$  de 20 y 40, presentan un comportamiento similar en ambos casos. Primeramente, se conserva una distancia segura en todo momento y el comando de aceleración resulta tener un comportamiento sin cambios bruscos, la diferencia principal con respecto al resto de los controladores es que al no llegar a la distancia límite, este controlador busca llegar a la velocidad establecida por el usuario de 20 [m/s]. De lo anterior, dado que el MIO va a una velocidad de 22 [m/s] el espacio entre ambos vehículos se acrecentó y, por tanto, el EGO obedece a la acción de seguir en su referencia y no busca llegar a la velocidad del vehículo MIO. Se puede observar como la distancia incluso es tal, que el rango del radar no es suficiente para cubrirla al final de la simulación.

Continuando con nuestro análisis comparativo del desempeño de los controladores se calculó el error RMSE con respecto a la velocidad de referencia, donde se obtuvo la Figura 8.2.6.7.

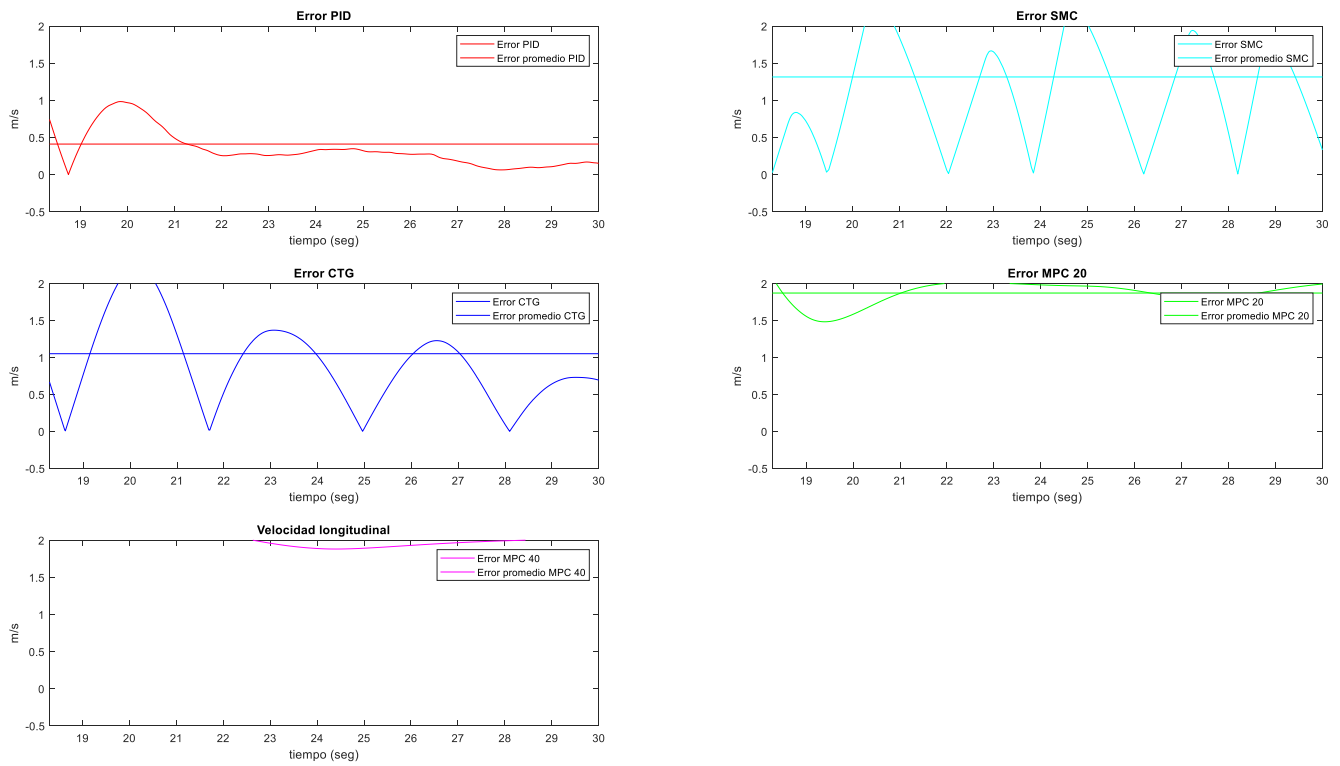


Figura 8.2.6.7. Variación del error RMSE de cada controlador con respecto a la velocidad de referencia

ESCENARIO 6									
Controlador	Vmax	%V	t1	$\tau$	te	RMSE	%RMSE	ta	vel
Clásico PI	22.98	4.45	18.84	14.40	21.9	0.414	2.30	4.3	
SMC	22.83	3.77	18.24	14.20	22.96	1.310	10.08	5.4	
CTG	24.24	10.18	18.58	14.50	23.14	1.050	8.08	5.3	
MPC 20	20.5	2.50	18.33	12.86	21.8	0.198	1.52	6.10	
MPC 40	19.44	-2.80	19.97	13.32	22.74	0.281	2.16	6.40	

Tabla 8.2.6.1. Comparación del rendimiento de los controladores diseñados, escenario 04.

De la Tabla 8.2.6.1, podemos notar que el principal diferenciador de los controladores es el porcentaje de sobre pico y el tiempo en que tarda en estabilizarse la señal, a mayor sobre pico, mayor cantidad de tiempo. En cuanto al resto de parámetros se observa que varían muy poco con diferencias de 1-2 [s]. Para clarificar y evaluar el desempeño de los controladores se realizó el índice de ponderación mostrado en la Tabla 8.2.6.2.



ESCENARIO 06									
Controlador	Vmax	%V	t1	$\tau$	te	RMSE	%RMSE vel	Puntuación total	Orden de desempeño
<b>Ponderación (0-1)</b>	1	1	3	3	5	5	5		
<b>Clásico PI</b>	22.98	4.45	56.52	43.20	109.50	2.07	11.51	250.23	<b>3</b>
<b>SMC</b>	22.83	3.77	54.72	42.60	114.80	6.55	50.38	295.66	<b>5</b>
<b>CTG</b>	24.24	10.18	55.74	43.50	115.70	5.25	40.38	295.00	<b>4</b>
<b>MPC 20</b>	20.50	2.50	54.99	38.58	109.00	0.99	7.62	234.18	<b>1</b>
<b>MPC 40</b>	19.44	-2.80	59.91	39.96	113.70	1.40	10.80	242.41	<b>2</b>

Tabla 8.2.6.2. Índice de ponderación para seleccionar el controlador con el mejor desempeño, escenario 06.

De la Tabla 8.2.6.2 se concluye que el controlador MPC se sobrepone ante el resto de controladores en este escenario particular, mientras que el SMC y el CTG tienen un desempeño muy similar entre sí; por otro lado, el PID se ajusta de una mejor manera que estos dos ante el cambio de referencia. A lo largo de las simulaciones realizadas en los diferentes escenarios se ha observado el comportamiento de cada controlador ante cada uno de ellos, de este análisis se determinó un valor de desempeño, el cual nos es de utilidad para concluir sobre los controladores propuestos en este estudio, lo anterior se resume en la Tabla 8.2.6.3.

Evaluación de Desempeño								
	Escenario 1	Escenario 2	Escenario 3_1	Escenario 3_2	Escenario 4	Escenario 5	Escenario 6	Total
<b>Clásico PID</b>	2	1	4	1	4	2	3	17
<b>SMC</b>	1	3	3	3	5	1	5	21
<b>CTG</b>	3	5	2	4	2	3	4	23
<b>MPC 20</b>	4	2	1	2	1	5	1	16
<b>MPC 40</b>	5	4	5	5	3	4	2	28

Tabla 8.2.6.3. Evaluación del desempeño de los cuatro controladores para cada escenario propuesto

Las conclusiones generales de esta evaluación de desempeño y de las simulaciones llevadas a cabo durante esta sección se abordan en el apartado nueve de este escrito.



## 9. Conclusión

Basado en los resultados de la sección anterior se concluyó que:

- Se identificaron cuatro maneras para implementar el controlador de crucero adaptativo: controlador clásico PID, SMC, CTG y MPC utilizando un modelo de primer orden para el vehículo ACC.
- El software utilizado durante este experimento, Matlab, es de suma utilidad para modificar y desarrollar los controladores propuestos, de tal forma que se pudo utilizar para simular los controladores propuestos con la ventaja de que ya contiene una interfaz integrada de antemano para el diseño de los mismos. También, se tiene la posibilidad de poder extrapolar el código y metodología utilizadas en pruebas físicas de validación si consideramos que el nivel actual de diseño del presente escrito es muy parecido a una metodología MIL.

Con respecto a la comparación de las estrategias de control se concluyó lo siguiente:

- El controlador clásico PID resulta bastante eficaz en este tipo de sistemas, ya que presenta estabilidad ante perturbaciones que ocurren en poco tiempo, de la misma manera ajusta su velocidad con el vehículo que pretende seguir (MIO) y por tanto mantiene una distancia segura en la mayoría de los casos.
- El controlador SMC y CTG presentaron un desempeño similar en la mayoría de los escenarios. Por un lado, el controlador SMC se adapta a los cambios con gran facilidad y velocidad de respuesta, siendo que su mayor desventaja es su comportamiento en sí, al tener mucha variación en poco tiempo en la señal de aceleración resulta mecánicamente no deseable para que el vehículo acelere. Contrario a lo anterior, el controlador CTG presenta un comportamiento bastante suave a lo largo de todas las simulaciones, con la gran desventaja de tener un tiempo respuesta más grande que el SMC.
- El controlador MPC inicialmente se planteó en dos variantes, el único parámetro modificado fue el horizonte de predicción, siendo que se modificó con un valor de veinte y cuarenta. De los seis escenarios se concluye que en la mayoría de los casos resulta más eficaz tener un horizonte de predicción menor, ya que el tiempo de respuesta mejora; sin embargo, esto hace que para situaciones donde la perturbación tiene una duración mayor y se presenta en periodo de tiempo largo y no de inmediato, el horizonte de predicción parece tener un mejor comportamiento.

- Tal y como se ha desarrollado a lo largo del escrito, cada controlador tiene sus respectivos aspectos positivos y negativos; sin embargo, el controlador MPC con un horizonte de predicción de veinte presenta el mejor comportamiento de manera general ante las distintas perturbaciones en los escenarios propuestos, este es el resultado esperado; no obstante, cabe resaltar que esto dependerá en gran medida de los parámetros elegidos. Vale la pena mencionar el hecho de que el controlador PID, SMC y CTG no presentan un comportamiento errático y su desempeño es parecido al del MPC, inclusive pareciera que su desempeño resulta mayor en algunos casos específicos. Lo anterior puede deberse a la variación de parámetros entre cada controlador, ya que cada estrategia se aproxima de manera diferente a la resolución del objetivo de control, también puede ser originado por la diferencia de apreciación en los valores comparados en el índice de ponderación asociado a cada escenario.

En general:

- Todos los controladores fueron evaluados y cumplieron el objetivo de control inicial, evitar una colisión entre vehículos, manteniendo una distancia segura y velocidad adecuada entre estos, siendo el MPC el que mejor desempeño mostró en las simulaciones se puede decir que es la estrategia de control recomendada para desarrollar un control de cruceo adaptativo.
- A lo largo del escrito se hizo uso de diversas herramientas para el diseño de los controladores, la mayoría de estas implementadas dentro del entorno de Matlab y Simulink mediante diagramas de bloque; por lo cual en lugar de elaborar nuevas herramientas se hizo uso las ya existentes; sin embargo, no se deja fuera la posibilidad de unir dichas herramientas con el uso de software o hardware externo especializado para el contexto de este trabajo de investigación, lo cual podría ser desarrollado como trabajo futuro.
- Adicionalmente, con el actual estado de desarrollo de las estrategias de control, se podría plantear una forma de validación para todas ellas mediante la fabricación de algún banco de pruebas físico; así como, un prototipo a escala del vehículo de interés con el propósito de realizar los mismos escenarios en un ambiente físico controlado. Alternativamente, como trabajo futuro se podrían implementar los resultados encontrados e inclusive mejorar el desempeño actual de los controladores mediante un refinado de las mismas estrategias de control.



# 10. Referencias

- [1] Agüero, H. (s. f.). *Historia del automóvil*. TINS Universidad Tecnológica de Perú.  
<http://www.librosmaravillosos.com/historiadelaautomovil/index.html>
- [2] *Historia de Tesla*. (s. f.). autobild.es. Recuperado 1 de diciembre de 2020, de  
<https://www.autobild.es/coches/tesla/historia>
- [3] Piedrafita, R. (1999). *Evolución histórica de la ingeniería de control* [Libro electrónico]. <http://automata.cps.unizar.es/regulacionautomatica/historia.PDF>
- [4] Colaboradores de Wikipedia. (2020, 23 septiembre). *Plataforma (informática)*. Wikipedia, la enciclopedia libre.  
[https://es.wikipedia.org/wiki/Plataforma\\_\(inform%C3%A1tica\)](https://es.wikipedia.org/wiki/Plataforma_(inform%C3%A1tica))
- [5] Bimbraw, K. (2015, 1 enero). *Autonomous Cars: Past, Present and Future - A Review of the Developments in the Last Century, the Present Scenario and the Expected Future of Autonomous Vehicle Technology*. Researchgate.  
[https://www.researchgate.net/publication/283757446\\_Autonomous\\_Cars\\_Past\\_Present\\_and\\_Future\\_-\\_A\\_Review\\_of\\_the\\_Developments\\_in\\_the\\_Last\\_Century\\_the\\_Present\\_Scenario\\_and\\_the\\_Expected\\_Future\\_of\\_Autonomous\\_Vehicle\\_Technology](https://www.researchgate.net/publication/283757446_Autonomous_Cars_Past_Present_and_Future_-_A_Review_of_the_Developments_in_the_Last_Century_the_Present_Scenario_and_the_Expected_Future_of_Autonomous_Vehicle_Technology)
- [6] Sjafrie, H. (2019). *Introduction to Self-Driving Vehicle Technology; Chapman & Hall/CRC Artificial Intelligence and Robotics*. CRC Press.
- [7] A History of Laser Scanning, Part 1: Space and Defence Applications
- [8] Taxonomy SAE. Definitions for terms related to driving automation systems for on-road motor vehicles. SAE Standard J, 3016, 2016.
- [9] Asadi, A., Tavana, M., Asadi, M., & Oliver, T. (2016, 29 agosto). *Autonomous Vehicles: Challenges, Opportunities and Future Implications for Transportation Policies*. Researchgate.  
[https://www.researchgate.net/publication/305904140\\_Autonomous\\_Vehicles\\_Challenges\\_Opportunities\\_and\\_Future\\_Implications\\_for\\_Transportation\\_Policies](https://www.researchgate.net/publication/305904140_Autonomous_Vehicles_Challenges_Opportunities_and_Future_Implications_for_Transportation_Policies)

- [10] Watzenig, D., & Horn, M. (2018). *Automated Driving: Safer and More Efficient Future Driving* (Softcover Reprint of the Original 1st 2017 ed.). Springer.
- [11] MRCagney & Transport and Economic Research Institute. (2017, diciembre). *Autonomous vehicles research report*. MRCagney. <https://www.mrcagney.com/uploads/case-studies/MRCAutonomousVehicles.pdf>
- [12] Texa Edu. (2017). *Advances Driver Assistance Systems Diagnosis and Calibration*. <https://www.texa.com/Upload/Depliant/d9c-en.pdf>
- [13] European Road Safety Observatory. (2016). *Advanced driver assistance systems 2016*. [https://ec.europa.eu/transport/road\\_safety/sites/roadsafety/files/ersosynthesis2016-adas15\\_en.pdf](https://ec.europa.eu/transport/road_safety/sites/roadsafety/files/ersosynthesis2016-adas15_en.pdf)
- [14] Wikipedia contributors. (2020, 30 noviembre). *Advanced driver-assistance systems*. Wikipedia. [https://en.wikipedia.org/wiki/Advanced\\_driver-assistance\\_systems](https://en.wikipedia.org/wiki/Advanced_driver-assistance_systems)
- [15] Kumar, V., Tunnell, J., Pasricha, S., & Bradley, T. (2018, 8 agosto). *Advanced Driver-Assistance Systems a path towards autonomous vehicles*. ResearchGate. [https://www.researchgate.net/publication/327253343\\_Advanced\\_Driver-Assistance\\_Systems\\_A\\_Path\\_Toward\\_Autonomous\\_Vehicles](https://www.researchgate.net/publication/327253343_Advanced_Driver-Assistance_Systems_A_Path_Toward_Autonomous_Vehicles)
- [16] Dugarry, A. (2004, junio). *Advanced Driver Assistance Systems Information Management and Presentation*. Cranfield University. <https://core.ac.uk/download/pdf/137892.pdf>
- [17] Rajamani, R. (2014). *Vehicle Dynamics and Control* (2nd 2012 ed.). Springer.
- [18] National Highway Traffic Safety Administration. (2014, julio). *Blind Spot Monitoring in Light Vehicles- System Performance*. [https://www.nhtsa.gov/DOT/NHTSA/NVS/Crash%20Avoidance/Technical%20Publications/2014/812045\\_Blind-Spot-Monitoring-in-Light-Vehicles-System-Performance.pdf](https://www.nhtsa.gov/DOT/NHTSA/NVS/Crash%20Avoidance/Technical%20Publications/2014/812045_Blind-Spot-Monitoring-in-Light-Vehicles-System-Performance.pdf)
- [19] Ogbac, S. (2020, 12 mayo). *Is Blind-Spot Monitoring Worth It? What Is It Exactly?* MotorTrend. <https://www.motortrend.com/news/blind-spot-monitoring/>
- [20] my-cardictionary. (s. f.-b). *Parking aid | Mein Autolexikon*. <https://www.my-cardictionary.com/driver-assistance-systems/parking-aid.html#:~:text=Parking%20aids%20are%20advanced%20driver,the%20driver%20about%20any%20obstacles>

- [21] BOSCH. (s. f.). *Parking aid*. <https://www.bosch-mobility-solutions.com/en/products-and-services/passenger-cars-and-light-commercial-vehicles/driver-assistance-systems/parking-aid/>
- [22] BOSCH. (s. f.-c). *Rear view system*. <https://www.bosch-mobility-solutions.com/en/products-and-services/passenger-cars-and-light-commercial-vehicles/driver-assistance-systems/rear-view-system/>
- [23] Knoll, P. (2010, abril). *Automotive Night Vision Systems – Status and Development Trends*.  
[https://www.researchgate.net/publication/281005005\\_Automotive\\_Night\\_Vision\\_Systems\\_-\\_Status\\_and\\_Development\\_Trends](https://www.researchgate.net/publication/281005005_Automotive_Night_Vision_Systems_-_Status_and_Development_Trends)
- [24] Oei, H. L., & Polak, P. H. (2002, diciembre). *Intelligent Speed Adaptation (ISA) and Road Safety*.  
[https://www.researchgate.net/publication/237799469\\_Intelligent\\_speed\\_adaptation\\_ISA\\_and\\_road\\_safety](https://www.researchgate.net/publication/237799469_Intelligent_speed_adaptation_ISA_and_road_safety)
- [25] Institute for Transport Studies, The University of Leeds, Carsten, O., & Tate, F. *INTELLIGENT SPEED ADAPTATION THE BEST COLLISION AVOIDANCE SYSTEM* (paper 324).  
<https://www-nrd.nhtsa.dot.gov/pdf/ESV/esv17/Proceed/00152.pdf>
- [26] Rosique, F., Navarro, P., Fernandez, C., & Padilla, A. (2019, febrero). *A Systematic Review of Perception System and Simulators for Autonomous Vehicles Research*. División de Sistemas e Ingeniería Electrónica (DSIE), Universidad Politécnica de Cartagena, Campus Muralla del Mar.  
[https://www.researchgate.net/publication/330900887\\_A\\_Systematic\\_Review\\_of\\_Perception\\_System\\_and\\_Simulators\\_for\\_Autonomous\\_Vehicles\\_Research](https://www.researchgate.net/publication/330900887_A_Systematic_Review_of_Perception_System_and_Simulators_for_Autonomous_Vehicles_Research)
- [27] Kvam, J. (2017). *Time of Flight: Principles, Challenges, and Performance* [Diapositivas]. st. [https://www.st.com/content/dam/technology-tour-2017/session-1\\_track-4\\_time-of-flight-technology.pdf](https://www.st.com/content/dam/technology-tour-2017/session-1_track-4_time-of-flight-technology.pdf)
- [28] Shinnosuke, H. & Department of Information Processing Interdisciplinary Graduate School of Science and Engineering Tokyo Institute of Technology. (2008, 30 junio). *Accurate measurement of distance and velocity using ultrasonic waves*. Acoustics.org.  
<https://acoustics.org/pressroom/httpdocs/155th/hirata.htm>
- [29] Pao, Charles (2019, 15 mayo). [Representación de los movimientos estimados por una IMU de nueve grados de libertad]. Optimize your IMU with Dynamic Calibration.  
<https://www.ceva-dsp.com/ourblog/optimize-your-imu-with-dynamic-calibration/>



[30] Briefs, T. (2020, 4 marzo). *IMU Contributions to Navigation and Safety for ADAS and Autonomous Vehicles*. Tech Briefs Media Group.

<https://www.techbriefs.com/component/content/article/tb/supplements/st/features/applications/36151>

[31] T. (2021, 24 junio). *ABS Revision*. Tim's Autotronics Autotronics Course at Unitec Northshore, NZ. <http://auto-tim.blogspot.com/2010/09/abs-revision.html>

[32] D. (2017, 26 septiembre). *¿Qué es un Odómetro o Topómetro? De Máquinas y Herramientas*. <https://www.demaquinasyherramientas.com/herramientas-de-medicion/odometro-tipos-y-usos>

[33] Bayern, M. (2019, 4 noviembre). *The top 3 companies in autonomous vehicles and self-driving cars*. ZDNet. <https://www.zdnet.com/article/the-top-3-companies-in-autonomous-vehicles-and-self-driving-cars/>

[34] *How Autonomous Vehicles Work | Let's Talk Autonomous Driving*. (s. f.). Let's Talk Autonomous Driving. Recuperado 29 de junio de 2021, de <https://ltad.com/about/how-autonomous-vehicles-work.html>

[35] Waymo. (2020, 25 marzo). *Designing the Waymo Driver* [Video]. YouTube. <https://www.youtube.com/watch?v=o8rCOKSDMcg>

[36] Cruise. (s. f.). Cruise. Recuperado 30 de junio de 2021, de <https://www.getcruise.com/technology/>

[37] BBC News. (2020, 22 enero). *GM's Cruise unveils its first driverless vehicle*. <https://www.bbc.com/news/technology-51171398>

[38] Argo AI. (2021, abril). *Argo, Developing a Self-Driving System You Can Trust*. <https://www.argo.ai/wp-content/uploads/2021/04/ArgoSafetyReport.pdf>

[39] Washington, DC: National Highway Traffic Safety Administration., Schnelle, S., Salaani, K., Barickman, J, & Elsasser, D (2019, diciembre). *Review of simulation frameworks and standards related to driving scenarios* ((Report No. DOT HS 812 815)).

[40] ARIZONA STATE UNIVERSITY, & Gopalakrishnan, V. (2018, noviembre). *A STUDY OF DRIVING SIMULATION PLATFORMS FOR AUTOMATED VEHICLES*. ARIZONA STATE UNIVERSITY.

[https://repository.asu.edu/attachments/218358/content/CAV\\_Final\\_Report\\_Vaishakh.pdf](https://repository.asu.edu/attachments/218358/content/CAV_Final_Report_Vaishakh.pdf)

[41] Changwoo, P, Seunghwan, C, & Hyeongcheol, L (2020, 11 abril). Vehicle-in-the-Loop in Global Coordinates for Advanced Driver Assistance System. *Applied Science MDPI*.

[42] embitel. (s. f.). *Handbook and tech guide: Kick-Start your functional testing project today! HIL, MIL, SIL* [Diapositivas]. embitel. <https://www.embitel.com/wp-content/uploads/Functional-Testing-Handbook-1.pdf>

[43] Xu, Z. (2017, 13 noviembre). *PaTAVTT: A Hardware-in-the-Loop Scaled Platform for Testing Autonomous Vehicle Trajectory Tracking*. Hindawi. <https://www.hindawi.com/journals/jat/2017/9203251/>

[44] Marthworks. (s. f.). *Automoción - Sistemas de conducción autónoma*. MATLAB & Simulink. Recuperado 21 de julio de 2021, de <https://es.mathworks.com/solutions/automated-driving.html>

[45] National Instrument. (2021, 4 mayo). *NI Acquires monoDrive to Accelerate Autonomous Vehicle Development*. NI. <https://www.ni.com/es-mx/about-ni/newsroom/news-releases/ni-acquires-monodrive-to-accelerate-autonomous-vehicle-developme.html>

[46] National Instrument. (s. f.). *monoDrive — monoDrive Simulator*. monoDrive. Recuperado 28 de julio de 2021, de <https://monodrive.readthedocs.io/en/latest/>

[47] CarSim. (s. f.). *CarSim Overview*. Recuperado 29 de julio de 2021, de <https://www.carsim.com/products/carsim/index.php>

[48] Team, C. (s. f.). *CARLA*. CARLA Simulator. Recuperado 4 de agosto de 2021, de <https://carla.org/>

[49] CARLA. (s. f.). *Introduction - CARLA Simulator*. CARLA. Recuperado 4 de agosto de 2021, de [https://carla.readthedocs.io/en/latest/start\\_introduction/#the-simulator](https://carla.readthedocs.io/en/latest/start_introduction/#the-simulator)

[50] LG Electronics America R&D Lab. (s. f.). *Home - SVL Simulator*. <https://www.svlsimulator.com/>. Recuperado 5 de agosto de 2021, de <https://www.svlsimulator.com/docs/>

[51] Wen, M.; Park, J.; Sung, Y.; Park, Y.W.; Cho, K. Virtual Scenario Simulation and Modeling Framework in Autonomous Driving Simulators. *Electronics* 2021, 10, 694. <https://doi.org/10.3390/electronics10060694>

[52] *SEGURIDAD FUNCIONAL AUTOMOTRIZ SEGÚN ISO 26262*. (s. f.). TÜV SÜD. Recuperado 10 de agosto de 2021, de <https://www.tuvsud.com/es->

mx/industrias/movilidad-e-industria-automotriz/automotriz-y-oem/seguridad-funcional-automotriz-segun-iso-26262

[53] Smith, C. (1978). *Tune to Win* (1st ed.). Carroll Smith Consulting.

[54] University of Luebeck [Prof. Georg Schildbach]. (2020, 3 mayo). *Vehicle Dynamics & Control - 09 Dynamic bicycle model with linear tires* [Vídeo]. YouTube.  
<https://www.youtube.com/watch?v=35IZIO6NrO0>

[55] MathWorks. (s. f.). *Adaptive Cruise Control with Sensor Fusion - MATLAB & Simulink - MathWorks Deutschland*. Recuperado 29 de septiembre de 2021, de <https://de.mathworks.com/help/driving/ug/adaptive-cruise-control-with-sensor-fusion.html>

[56] Eyisi, Emeka & Zhang, Zhenkai & Koutsoukos, Xenofon & Porter, Joseph & Karsai, Gabor & Sztipanovits, Janos. (2013). Model-Based Control Design and Integration of Cyberphysical Systems: An Adaptive Cruise Control Case Study. *Journal of Control Science and Engineering*. 2013. 10.1155/2013/678016.

[57] Winner, H., Prokop, G., & Maurer, M. (2019). *Automotive Systems Engineering II* (Softcover Reprint of the Original 1st 2018 ed.). Springer.

[58] Ali, Zeeshan. (2011). Transitional controller design for adaptive cruise control systems.

[59] Farrell, J., & Plycarpou, M. (2006). *Chapter 5.4.2 - Sliding Mode Control | Engineering360*. Engineering360.  
<https://www.globalspec.com/reference/21394/160210/Chapter-5-4-2-Sliding-Mode-Control#:~:text=The%20manifold%20described%20by%20s%20%E2%80%93%200%20is,the%20variable%20s%20to%20zero%20in%20finite%20time.>

[60] MathWorks. (s. f.-b). *Simulate adaptive cruise control using model predictive controller - Simulink - MathWorks Deutschland*. Recuperado 29 de septiembre de 2021, de <https://de.mathworks.com/help/mpc/ref/adaptivecruisecontrolsystem.html>

[61] Mathworks. (s. f.). *Highway Lane Following - MATLAB & Simulink*. Recuperado 30 de septiembre de 2021, de <https://www.mathworks.com/help/mpc/ug/highway-lane-following.html>

[62] ISO 15622, Transport information and control systems — Adaptive Cruise Control Systems — Performance requirements and test procedures. Recuperado 05 de octubre de 2021, de [http://www.puntofocal.gov.ar/notific\\_otros\\_miembros/sau283\\_t.pdf](http://www.puntofocal.gov.ar/notific_otros_miembros/sau283_t.pdf)

[63] ISO 22178, Intelligent transport systems — Low Speed Following (LSF) Systems — Performance requirements and test procedures. Recuperado 05 de octubre de 2021, de <https://pdfslide.tips/reader/f/iso-22178-2009>