

Proyecto Fin de Máster

Ingeniería en Electrónica, Automática y Robótica

Desarrollo de métodos para la estimación del ángulo de deslizamiento lateral en vehículos

Autor: Adrián Antonio Quijije Hernández

Tutor: Carlos Bordons Alba

Dpto. de Ingeniería Electrónica
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla

Sevilla, 2020



Proyecto Fin de Máster
Ingeniería en Electrónica, Automática y Robótica

Desarrollo de métodos para la estimación del ángulo de deslizamiento lateral en vehículos

Autor:

Adrián Antonio Quijije Hernández

Tutor:

Carlos Bordons Alba

Catedrático

Dpto. de Ingeniería de Sistemas y Automática

Escuela Técnica Superior de Ingeniería

Universidad de Sevilla

Sevilla, 2020

Proyecto Fin de Máster: Desarrollo de métodos para la estimación del ángulo de deslizamiento lateral en vehículos

Autor: Adrián Antonio Quijije Hernández

Tutor: Carlos Bordons Alba

El tribunal nombrado para juzgar el Proyecto arriba indicado, compuesto por los siguientes miembros:

Presidente:

Vocales:

Secretario:

Acuerdan otorgarle la calificación de:

Sevilla, 2020

El Secretario del Tribunal

A mi familia

A mis amigos

A mis profesores

A mis compañeros

Agradecimiento

Este proyecto supone el final de esta etapa de estudios y desarrollo personal cargado de experiencias y de personas para recordar.

En primer lugar, me gustaría agradecer el cariño y ánimo de mis padres y mi hermana que a la distancia me dieron apoyo.

En segundo lugar, a mis compañeros y amigos, con los que compartí buenas experiencias.

Por último y no menos importante, me gustaría agradecer a mi tutor, Carlos Bordons Alba, por la oportunidad para realizar este proyecto.

Resumen

Uno de los mayores obstáculos para implementar los sistemas de control de estabilidad en los vehículos, es disponer de toda la información que pueda influir en los datos de salida del sistema, uno de ellos es el ángulo de deslizamiento lateral (β) de los vehículos.

Es muy importante conocer en cada instante de tiempo el ángulo de deslizamiento lateral del vehículo para poder mejorar e implementar nuevos sistemas de control de estabilidad del automóvil. El ángulo β , al depender de muchísimos parámetros, necesita sistemas bastante complejos y caros para ser obtenido lo que impide que sea implantado en los vehículos de producción.

En este trabajo se realizarán algunos métodos para poder estimarlos, compararlos con datos de simulación como Carsim y compararlos con datos de experimentos reales del vehículo FOX de la Universidad de Sevilla.

Abstract

One of the biggest obstacles to implementing stability control systems in vehicles is having all the information that can influence the output data of the system, one of them is the slip angle (Beta) of the vehicles.

It is very important to know the vehicle's slip angle at every moment in time in order to improve and implement new vehicle stability control systems. The beta angle, as it depends on many parameters, requires quite complex and expensive systems to be obtained, which prevents it from being implemented in production vehicles.

In this work, some methods will be carried out to be able to estimate them, compare them with simulation data such as Carsim and compare them with data from real experiments of the FOX vehicle of the University of Seville.

Agradecimiento	ix
Resumen	xi
Abstract	xiii
Índice	xv
Índice de Tablas	xvii
Índice de Figuras	xix
Notación	xxiii
1 Introducción	2
1.1 <i>Introducción</i>	2
1.2 <i>Objetivos</i>	3
2 Ángulo de deslizamiento lateral	5
2.1 <i>Introducción</i>	5
2.1.1 <i>Definición</i>	6
2.1.2 <i>Parámetros de los que depende</i>	9
2.2 <i>Modelo lineal del vehículo</i>	9
3 Descripción de los métodos de estimación	11
3.1 <i>Introducción</i>	11
3.2 <i>Estimación basada en observadores</i>	11
3.3 <i>Redes Neuronales</i>	13
3.4 <i>Patentes de fabricantes</i>	14
4 Diseño y validación de los estimadores	15
4.1 <i>Introducción</i>	15
4.2 <i>Observador Luenberger</i>	15
4.3 <i>Filtros de Kalman</i>	16
4.4 <i>Redes Neuronales</i>	19
4.4.1 <i>Introducción</i>	19
4.4.2 <i>Redes neuronales artificiales</i>	19
4.4.3 <i>Método usado para la estimación</i>	21
5 Modelos de estimación del ángulo de deslizamiento lateral	23
5.1 <i>Introducción</i>	23
5.2 <i>Características del vehículo</i>	23
5.3 <i>Maniobras de las pruebas para simular</i>	27
5.3.1 <i>Giro circular a 40 km/h</i>	28
5.3.2 <i>Doble cambio de carril a 120 km/h</i>	28
5.3.3 <i>Movimiento sinusoidal a 50 km/h</i>	29
5.4 <i>Modelo Lineal</i>	30
5.4.1 <i>Simulaciones</i>	31
5.5 <i>Modelo basado en Observador Luenberger</i>	37
5.5.1 <i>Simulaciones</i>	39
5.6 <i>Modelo basado en Filtro de Kalman</i>	44

5.6.1	Simulaciones	45
5.7	<i>Modelo basado en Redes Neuronales</i>	50
5.7.1	Experimento	51
	<i>Resumen</i>	54
6	Experimento con datos del vehículo FOX	55
6.1	<i>Introducción</i>	55
6.2	<i>Parámetros del FOX</i>	56
6.3	<i>Prueba con el modelo Lineal</i>	58
6.4	<i>Prueba con el modelo basado en Observador Luenberger</i>	59
6.5	<i>Prueba con el modelo basado en Filtro de Kalman</i>	60
6.6	<i>Prueba con la red neuronal</i>	61
7	Conclusiones	63
7.1	<i>Conclusiones</i>	63
	Referencias	64
	Anexos	66
	<i>Parámetros del vehículo Class-B</i>	66
	<i>Matrices del vehículo class-B</i>	67
	<i>Presentación de resultados</i>	68
	<i>Parámetros del vehículo FOX</i>	70
	<i>Matrices del vehículo FOX</i>	71
	<i>Presentación de resultados</i>	71
	<i>Red Neuronal</i>	73

ÍNDICE DE TABLAS

Tabla 5–1 Parámetros del vehículo Class-B de simulación	27
Tabla 5–2 Error medio en la estimación del ángulo de deslizamiento lateral	54
Tabla 5–3 Error máximo en la estimación del ángulo de deslizamiento lateral	54
Tabla 6–1 Parámetros del vehículo FOX	56

ÍNDICE DE FIGURAS

Figura 1-1: Funcionamiento del control de estabilidad.	3
Figura 2-1: Sistema coordenadas vehículo.	5
Figura 2-2: Sistema coordenadas vehículo.	6
Figura 2-3: Esquema de signo del ángulo de deslizamiento lateral.	6
Figura 2-4: Sistema plano del vehículo.	7
Figura 2-5: Ángulo de deslizamiento lateral de un neumático.	8
Figura 2-6: Relación entre el ángulo de deslizamiento lateral de un neumático y las fuerzas laterales.	8
Figura 2-7: Modelo de bicicleta	9
Figura 3-1: Esquema de estudio [6]	11
Figura 3-2: Esquema de estudio [7]	12
Figura 3-3: Esquema de estudio [15]	13
Figura 3-4: Esquema de alto nivel del proceso de estimación de redes neuronales artificiales (ANN)	13
Figura 3-5: Red neuronal de [5].	14
Figura 4-1: Estructura del Observador	15
Figura 4-2: Esquema de estados del Observador Luenberger	16
Figura 4-3: Etapas del Filtro de Kalman.	17
Figura 4-4: Esquema de funcionamiento de una neurona	19
Figura 4-5: Esquema de la arquitectura de una red	21
Figura 4-6: Esquema de alto nivel	21
Figura 4-7: Estructura de red neuronal	22
Figura 5-1: Vehículo seleccionado	23
Figura 5-2: Vehículo seleccionado	24
Figura 5-3: Información del eje delantero	24
Figura 5-4: Información del eje trasero	25
Figura 5-5: Información de los neumáticos	25
Figura 5-6: Información del eje trasero	26
Figura 5-7: Información de los neumáticos	26
Figura 5-8: Maniobras en CarSim	27
Figura 5-9: Ángulo de giro, simulación giro circular 40 km/h. Modelo lineal	28
Figura 5-10: Ángulo de giro, simulación doble cambio de carril a 120 km/h. Modelo lineal.	28
Figura 5-11: Ángulo de giro, movimiento sinusoidal. Modelo lineal	29
Figura 5-12: Sistema lineal Simulink	30
Figura 5-13: Bloque sistema lineal	31
Figura 5-14: Ángulo de deslizamiento lateral, simulación giro circular 40 km/h. Modelo lineal	31

Figura 5-15: Ángulo de guiñada, simulación giro circular 40 km/h. Modelo lineal	32
Figura 5-16: Aceleración lateral, simulación giro circular 40 km/h. Modelo lineal	32
Figura 5-17: Error β Carsim respecto β de la simulación simulación giro circular 40 km/h.	33
Figura 5-18: Ángulo de desplazamiento lateral, simulación doble cambio de carril a 120 km/h. Modelo lineal.	33
Figura 5-19: Ángulo de guiñada, simulación doble cambio de carril a 120 km/h. Modelo lineal.	34
Figura 5-20: Aceleración lateral, simulación doble cambio de carril a 120 km/h. Modelo lineal.	34
Figura 5-21: Error β Carsim respecto β de la simulación doble cambio de carril a 120 km/h.	35
Figura 5-22: Ángulo de desplazamiento lateral, movimiento sinusoidal. Modelo lineal	35
Figura 5-23: Ángulo de guiñada, movimiento sinusoidal. Modelo lineal	36
Figura 5-24: Aceleración lateral, movimiento sinusoidal. Modelo lineal	36
Figura 5-25: Error β Carsim respecto β de la simulación movimiento sinusoidal. Modelo lineal.	37
Figura 5-26: Sistema con Observador Luenberger	38
Figura 5-27: Bloque de Observador Luenberger	38
Figura 5-28: Ángulo de deslizamiento lateral, simulación giro circular 40 km/h. basado en Observador Luenberger.	Modelo 39
Figura 5-29: Ángulo de guiñada, simulación giro circular 40 km/h. basado en Observador Luenberger.	Modelo 39
Figura 5-30: Error β Carsim respecto β de la simulación simulación giro circular 40 km/h. en Observador Luenberger.	Modelo basado 40
Figura 5-31: Ángulo de desplazamiento lateral, simulación doble cambio de carril a 120 km/h. basado en Observador Luenberger.	Modelo 40
Figura 5-32: Ángulo de guiñada, simulación doble cambio de carril a 120 km/h. basado en Observador Luenberger.	Modelo 41
Figura 5-33: Error β Carsim respecto β de la simulación doble cambio de carril a 120 km/h. basado en Observador Luenberger.	Modelo 41
Figura 5-34: Ángulo de desplazamiento lateral, simulación movimiento sinusoidal a 50 km/h. basado en Observador Luenberger.	Modelo 42
Figura 5-35: Ángulo de guiñada, simulación movimiento sinusoidal a 50 km/h. basado en Observador Luenberger.	Modelo 42
Figura 5-36: Error β Carsim respecto β de la simulación movimiento sinusoidal a 50 km/h. Modelo basado en Observador Luenberger.	43
Figura 5-37: Sistema lineal Simulink	44
Figura 5-38: Configuración filtro de Kalman	44
Figura 5-39: Ángulo de desplazamiento lateral, simulación giro circular 40 km/h. Modelo Filtro de Kalman.	45
Figura 5-40: Ángulo de guiñada, simulación giro circular 40 km/h. Modelo Filtro de Kalman.	45
Figura 5-41: Error β Carsim respecto β de la simulación giro circular 40 km/h. Modelo Filtro de Kalman.	46
Figura 5-42: Ángulo de desplazamiento lateral, simulación doble cambio de carril a 120 km/h. Modelo Filtro de Kalman.	46
Figura 5-43: Ángulo de guiñada, simulación doble cambio de carril a 120 km/h. Modelo Filtro de Kalman.	47
Figura 5-44: Error β Carsim respecto β de la simulación doble cambio de carril a 120 km/h. Modelo Filtro de	

Kalman.	47
Figura 5-45: Ángulo de desplazamiento lateral, movimiento sinusoidal. a 50km/h. Filtro de Kalman.	48
Figura 5-46: Ángulo de guiñada, movimiento sinusoidal. a 50km/h. Filtro de Kalman.	48
Figura 5-47: Error β Carsim respecto β de la simulación movimiento sinusoidal 50 km/h. Modelo Filtro de Kalman.	49
Figura 5-48: Librerías de la red neuronal.	50
Figura 5-49: Red neuronal implementada para la estimación del ángulo de deslizamiento lateral.	50
Figura 5-50: Entrenamiento de la red neuronal.	51
Figura 5-51: Estimación de β de la red neuronal a la prueba de doble cambio de carril a 120km/h.	51
Figura 5-52: Error entre el β de CarSim y el β a la salida de la red neuronal.	52
Figura 5-53: Entrenamiento de la red neuronal.	52
Figura 5-54: Estimación de β de la red neuronal a la prueba de doble cambio de carril a 120 km/h.	53
Figura 5-55: Error entre el β de CarSim y el β a la salida de la red neuronal.	53
Figura 6-1: Vehículo FOX.	55
Figura 6-2: Trayectoria, prueba con FOX.	56
Figura 6-3: Ángulo de giro, prueba con FOX.	57
Figura 6-4: Velocidad longitudinal del vehículo FOX.	57
Figura 6-5: Ángulo de deslizamiento lateral, con modelo lineal.	58
Figura 6-6: Error entre β real y β estimado con modeo lineal.	58
Figura 6-7: Ángulo de deslizamiento lateral, con modelo basado en observador Luenberger.	59
Figura 6-8: Error entre β real y β estimado con modeo basado en observador Luenberger.	59
Figura 6-9: Ángulo de deslizamiento lateral, con modelo basado en filtro de Kalman.	60
Figura 6-10: Error entre β real y β estimado con modeo basado en filtro de Kalman	60
Figura 6-11: Entrenamiento de la red neuronal.	61
Figura 6-12: Estimación de β de la red neuronal.	61
Figura 6-13: Error entre la β estimada y la β de la prueba del vehículo FOX.	62

α	Ángulo de deslizamiento de la rueda	rad
β	Ángulo de deslizamiento lateral del vehículo	rad
δ	Ángulo de giro del volante	rad
V_x	Velocidad longitudinal, sobre la componente x	m/s
V_y	Velocidad lateral, sobre la componente y	m/s
Γ	Ángulo entre sistema de coordenadas global y vector velocidad	rad
Ψ	Ángulo entre el sistema de coordenadas global y eje longitudinal del vehículo	rad
C_y	Curva de rigidez del neumático	-
F_y	Fuerza lateral, sobre la componente y	N
F_x	Fuerza longitudinal, sobre la componente x	N
μ	Coefficiente de rozamiento	-
g	Aceleración de gravedad	m/s ²
I_z	Momento de inercia del vehículo alrededor del eje z	kgm ²
m	Masa del vehículo	kg
l_f	Distancia entre el centro de gravedad y el eje delantero	m
l_r	Distancia entre el centro de gravedad y el eje trasero	m
M_z	Momento corrector de velocidad angular	Nm
S	Área frontal del vehículo	m ²
C_{af}	Rigidez a deslizamiento lateral en la rueda delantera	N/rad
C_{ar}	Rigidez a deslizamiento lateral en la rueda trasera	N/rad

1 INTRODUCCIÓN

La ciencia no es sino una perversión de sí misma, a menos que tenga como objetivo final el mejoramiento de la humanidad.

- Nikola Tesla -

1.1 Introducción

En el último siglo el vehículo se ha convertido en el medio de transporte mas importante. El aumento de automóviles en circulación viene acompañado de algunos problemas como el impacto medioambiental debido a las emisiones de gases a la atmósfera, y los accidentes de tráfico hasta llegar a convertirse en una de las primeras causas de muerte en el mundo.

En la actualidad los automóviles cuentan con un gran número de sistemas de seguridad entre los que destacan los sistemas de control de estabilidad, que son capaces de corregir los errores cometidos por el conductor modificando la trayectoria del vehículo de manera autónoma. Debido a su complejidad, el correcto funcionamiento de estos sistemas implica entender en profundidad la dinámica del vehículo para poder recoger los parámetros necesarios para mejorar la estabilidad en todo tipo de situaciones.

Los vehículos con avances tecnológicos tienen muchos más sensores, lo que permite desarrollos de algunos sistemas de control como es el caso de estabilidad. Estos sistemas necesitan una medición precisa de las señales, ya que una medición errónea por parte del sensor puede ocasionar una mala decisión o perdida del control.

Uno de los parámetros clave para entender el comportamiento lateral del automóvil es el ángulo de deslizamiento lateral o ángulo β , que es el que forma el eje longitudinal del vehículo con su dirección de desplazamiento.

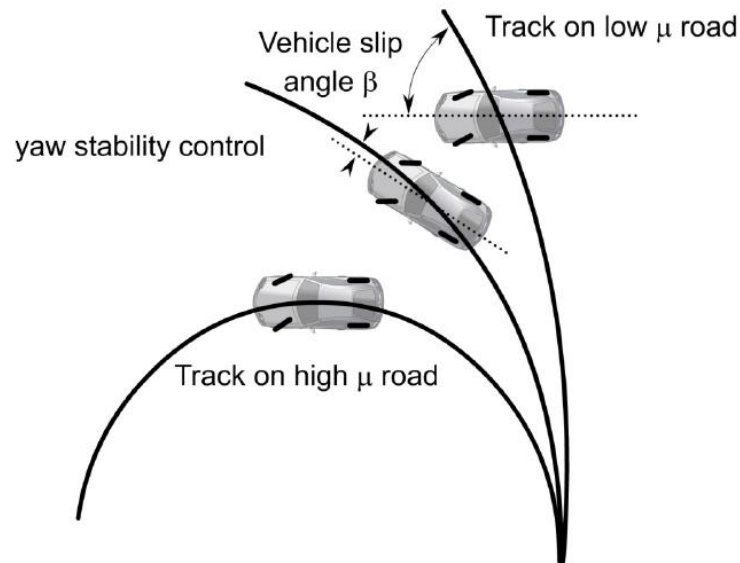


Figura 1-1: Funcionamiento del control de estabilidad.

La dificultad de estimar este parámetro ha dado lugar a muchas investigaciones de diversos autores sobre métodos para su cálculo, y en este proyecto se propondrá recopilar y comparar algunos métodos de estimación del ángulo de deslizamiento lateral.

1.2 Objetivos

Entre los objetivos de este trabajo de fin de máster está, primero entender el comportamiento lateral del vehículo profundizando en el fenómeno de deriva, estudiando el origen de éste y analizar las variables de las que depende. Y, segundo, revisar las técnicas empleadas por distintos autores para la estimación del parámetro β , y luego realizar la simulación de algunos métodos y su comparación entre sí.

2 ÁNGULO DE DESLIZAMIENTO LATERAL

2.1 Introducción

En el vehículo al realizarse algunas maniobras de giro se producen fuerzas laterales sobre los neumáticos que modifican su trayectoria [1]. Además, existen algunos factores que afectan a la dinámica lateral de un vehículo:

- Dimensionales: Distancia entre ejes del vehículo, la vía por la que circula el vehículo.
- Masas: Posición del centro de gravedad, relación masa suspendida/ no suspendida.
- Aerodinámicas: coeficientes aerodinámicos de fuerza lateral y momento de guiñada.
- Dinámicos: ángulo de balanceo y cabeceo, velocidad de guiñada, ángulo de deslizamiento lateral.
- Neumáticos: Características laterales y esfuerzos transversales.
- Suspensión: características geométricas y dinámicas.
- Factores externos o características del medio: el ángulo de incidencia del aire, las irregularidades de la calzada, el radio de curva y peralte de la calzada y el coeficiente de adherencia.
- Velocidad: si existe un valor crítico el vehículo puede presentar comportamientos inestables.

Conocer estos parámetros en cada instante de tiempo es muy importante para controlar la estabilidad del vehículo. Entre ellos uno de los más importantes es el ángulo de deslizamiento lateral para evaluar la estabilidad del vehículo durante las maniobras. En la figura 2-1 se describe el sistema de coordenadas del vehículo.

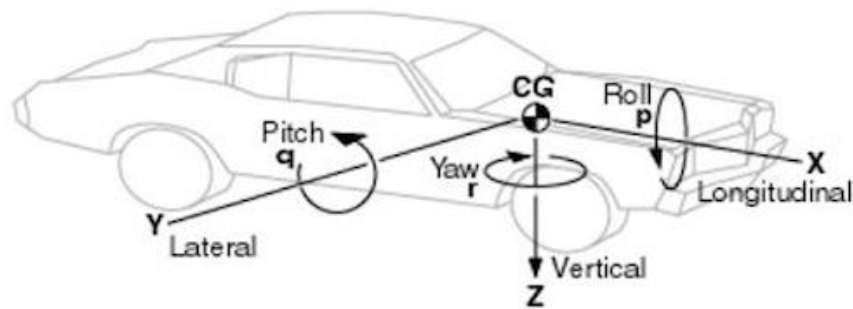


Figura 2-1: Sistema coordenadas vehículo.

donde:

- Eje x: Dirección longitudinal
- Eje y: Dirección transversal
- Eje z: Dirección vertical
- Momento p: Balance
- Momento q: Cabeceo
- Momento r: Guiñada

2.1.1 Definición

Un vehículo al desplazarse en una trayectoria rectilínea la dirección del centro de gravedad y el eje x del vehículo coinciden, cuando está tomando una curva la orientación del vehículo se modifica y provoca que el eje x y el vector velocidad (dirección real hacia la que se desplaza el vehículo) dejen de coincidir. El ángulo de deslizamiento lateral se forma entre la dirección del vehículo (eje x) y el vector velocidad que se descompone en velocidad longitudinal (V_x) y velocidad lateral (V_y). El ángulo de deslizamiento lateral se representa con la letra “ β ”. En la figura 2-2 se muestra las fuerzas que actúan sobre el centro de gravedad, los neumáticos y el sistema de coordenadas del vehículo.

El signo del ángulo de deslizamiento lateral se describe en la figura 2-3, es positivo cuando se gira en sentido horario y es negativo cuando gira en sentido antihorario. [2]

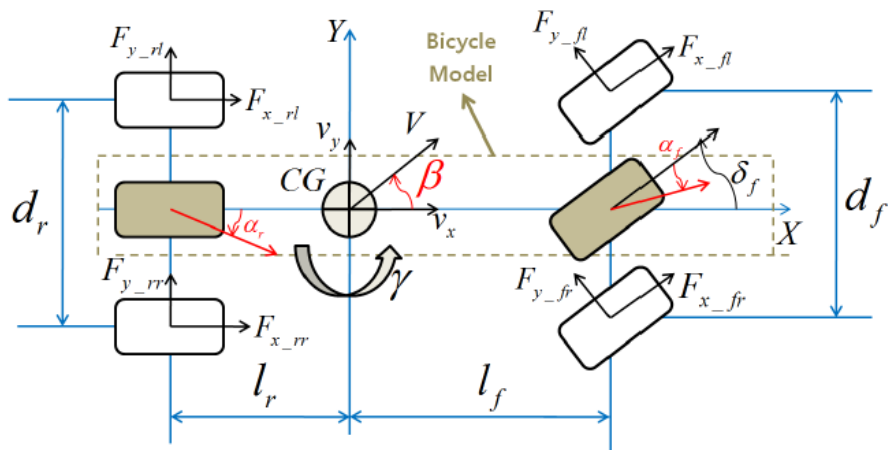


Figura 2-2: Sistema coordenadas vehículo.

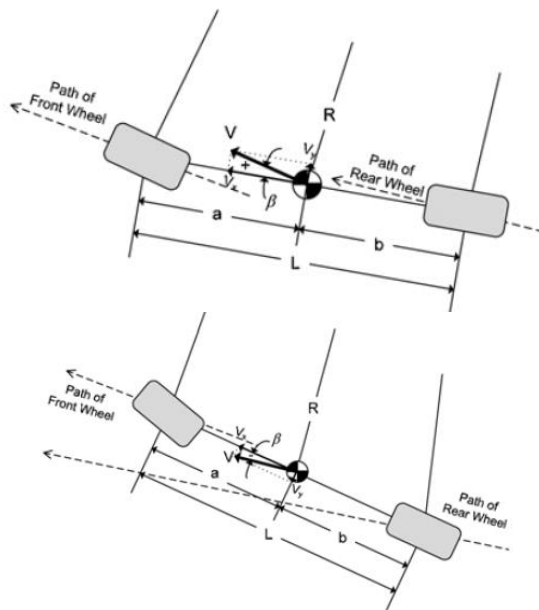


Figura 2-3: Esquema de signo del ángulo de deslizamiento lateral.

Se puede obtener el ángulo β con las componentes del vector velocidad con la fórmula 2-1:

$$\beta = \tan^{-1} \left(\frac{V_y}{V_x} \right) \quad (2-1)$$

donde:

- β : Ángulo de deslizamiento lateral
- V_x : Velocidad longitudinal, sobre la componente x
- V_y : Velocidad lateral, sobre la componente y

Desde el centro de gravedad del vehículo en el sistema plano del vehículo, descrito en la figura 2-2, se puede describir el ángulo β como la diferencia entre la dirección de la velocidad (γ), que es el ángulo formado entre el sistema de coordenadas globales y el vector velocidad, y el ángulo de guiñada (Ψ), formado entre el sistema de coordenadas globales y el eje longitudinal del vehículo.

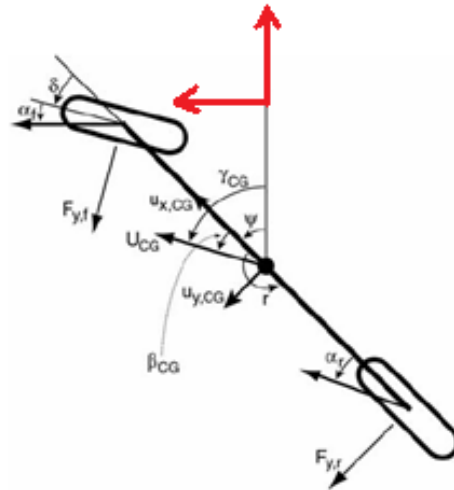


Figura 2-4: Sistema plano del vehículo.

$$\beta = \gamma - \Psi \quad (2-2)$$

donde:

- β : Ángulo de deslizamiento lateral
- γ : Ángulo entre sistema de coordenadas global y vector velocidad
- Ψ : Ángulo entre el sistema de coordenadas global y eje longitudinal del vehículo.

Los neumáticos proporcionan el único contacto del vehículo con la carretera, desarrollan fuerzas que dan como resultado la aceleración lateral. Cuando se da un cambio de dirección en el vehículo, las huellas sucesivas de los neumáticos que están en contacto con la carretera se desplazan lateralmente respecto con las huellas que ya están en contacto con la carretera. Con esto, se crea un ángulo de deslizamiento del neumático, que proporciona una estimación de la rigidez al deslizamiento lateral en la rueda. También se puede definir como la relación entre la velocidad lateral y el avance de la rueda. [2]

El ángulo entre la línea media de los neumáticos en posición original y la localización nueva se le llama ángulo de deslizamiento de los neumáticos, denominado α . En la figura 2-5 se muestra el ángulo de deslizamiento en relación a la fuerza lateral del neumático.

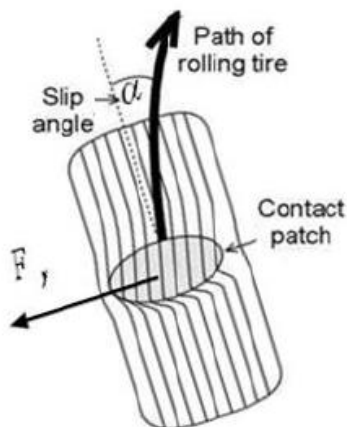


Figura 2-5: Ángulo de deslizamiento lateral de un neumático.

Durante una conducción normal, las fuerzas laterales generadas por los neumáticos son proporcionales al ángulo de deslizamiento lateral de los neumáticos, siempre y cuando este no exceda de 5 grados.

La expresión que define tal relación es:

$$F_y = C_y * \alpha \quad (2-3)$$

Donde:

- C_y : es la curva de rigidez del neumático.
- F_y : es la fuerza lateral, sobre la componente y.
- α : ángulo de deslizamiento lateral de los neumáticos.

En la figura 2-6 se muestra la relación que existe entre la fuerza lateral y el ángulo de deslizamiento de los neumáticos. Para ángulos de deslizamiento pequeños las fuerzas aumentan linealmente, lo que se llama región lineal, mientras que, para ángulos de deslizamiento más grandes, en la región no lineal, las fuerzas se saturan.

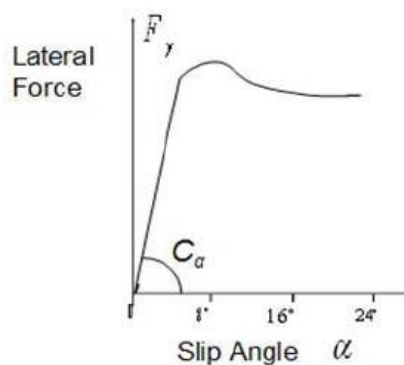


Figura 2-6: Relación entre el ángulo de deslizamiento lateral de un neumático y las fuerzas laterales.

2.1.2 Parámetros de los que depende

Es bastante importante conocer en cada instante de tiempo el ángulo de deslizamiento lateral del vehículo para poder realizar diversos sistemas de control de estabilidad del vehículo.

- Aceleración lateral: Es uno de los parámetros que más influye en la estimación lateral del vehículo. a baja velocidad del vehículo, la aceleración lateral es despreciable y β es pequeño. En cambio a altas velocidades, la aceleración lateral del vehículo es mayor y β aumenta considerablemente.
- Velocidad longitudinal: Normalmente se calcula a partir de la velocidad angular de las ruedas, este valor no ofrece un valor real cuando existe un elevado deslizamiento, ya que una fuerte aceleración o deceleración, las ruedas se bloquean.
- Dirección: Es la dirección marcada por el conductor y que debe seguir el vehículo.
- Velocidad de guiñada: Velocidad angular de rotación sobre el eje z del vehículo, que aparece al girar las ruedas directrices para realizar una curva.
- Características de los neumáticos: Curva de rigidez, presión, orientación, dimensiones, etc.
- Condiciones de la carretera: influyen en la velocidad lateral y longitudinal respectivamente.
 - Peralte: pendiente transversal de la calzada.
 - Inclinación: pendiente longitudinal de la calzada.
- Propiedades de interfaz entre neumático y calzada: fuerzas de adherencia no lineales.
- Características del vehículo:
 - Ángulo de balanceo
 - Ángulo de cabeceo
 - Carga del vehículo
- Ángulo del volante y del par corrector

2.2 Modelo lineal del vehículo

En los estudios de estabilidad y maniobrabilidad de vehículos se utilizan comúnmente un modelo dinámico simplificado de dos grados de libertad, este es conocido como el modelo de la bicicleta [3] [4]. En este diseño se centraliza el efecto producido sobre las dos ruedas traseras y las dos ruedas delanteras del vehículo, una única rueda delantera y una trasera como muestra la figura 5-1.

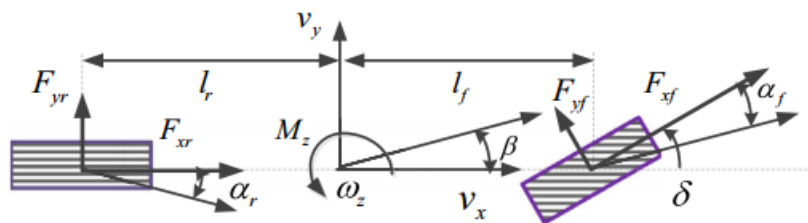


Figura 2-7: Modelo de bicicleta

Se aproximan las funciones trigonométricas para ángulos pequeños, no se consideran los efectos dinámicos como el cabeceo; y tampoco las variaciones de las fuerzas normales de las ruedas. Para la reproducción de las fuerzas laterales en los neumáticos se usa un modelo lineal, considerando valores de C_α constantes. Este modelo de la bicicleta normalmente se representa con las ecuaciones 5-1.

$$\begin{cases} m\dot{\beta} = -m\omega_z + \frac{1}{v_x}(F_{yf} + F_{yr}) \\ I_z\dot{\omega}_z = F_{yf}l_f - F_{yr}l_r \end{cases} \quad (2-4)$$

Donde:

$$F_{yf} = 2 * \alpha_f * C_{\alpha f} * \cos(\delta) \quad (2-5)$$

$$F_{yr} = -2 * \alpha_r * C_{\alpha r}$$

$$\alpha_f = \delta - \frac{\omega_z l_f}{V_x} - \beta$$

$$\alpha_r = \frac{\omega_z l_r}{V_x} - \beta$$

Reemplazando 2-5 en 2-4, se puede escribir como:

$$\begin{cases} \dot{x} = Ax + B\delta \\ y = Cx \end{cases} \quad (2-6)$$

$$\begin{bmatrix} \dot{\beta} \\ \dot{\gamma} \end{bmatrix} = \begin{bmatrix} 2 * \frac{-C_{\alpha f} - C_{\alpha r}}{m * V_x} & 2 * \frac{-l_f * C_{\alpha f} + l_r * C_{\alpha r}}{m * V_x^2} - 1 \\ 2 * \frac{-l_f * C_{\alpha f} + l_r * C_{\alpha r}}{I_{zz}} & 2 * \frac{-l_f^2 * C_{\alpha f} - l_r^2 * C_{\alpha r}}{I_{zz} * V_x} \end{bmatrix} \begin{bmatrix} \beta \\ \gamma \end{bmatrix} + \begin{bmatrix} \frac{2 * C_{\alpha f}}{m * V_x} \\ \frac{2 * l_f * C_{\alpha f}}{I_{zz}} \end{bmatrix} \delta$$

Donde:

$$A = \begin{bmatrix} 2 * \frac{-C_{\alpha f} - C_{\alpha r}}{m * V_x} & 2 * \frac{-l_f * C_{\alpha f} + l_r * C_{\alpha r}}{m * V_x^2} - 1 \\ 2 * \frac{-l_f * C_{\alpha f} + l_r * C_{\alpha r}}{I_{zz}} & 2 * \frac{-l_f^2 * C_{\alpha f} - l_r^2 * C_{\alpha r}}{I_{zz} * V_x} \end{bmatrix}$$

$$B = \begin{bmatrix} \frac{2 * C_{\alpha f}}{m * V_x} \\ \frac{2 * l_f * C_{\alpha f}}{I_{zz}} \end{bmatrix}$$

$$C = [0 \quad 1]$$

Donde:

- β : ángulo de deslizamiento lateral.
- ω_z : velocidad de guiñada.
- I_{ZZ} : momento de inercia respecto al eje vertical.
- l_f y l_r : distancias de los ejes delantero y trasero desde el centro de gravedad.
- $C_{\alpha r}$ y $C_{\alpha f}$: coeficientes de rigidez de la rueda delantera y trasera respectivamente.
- δ : ángulo de dirección de las ruedas delanteras (promedio).
- V_x : velocidad longitudinal.
- $\dot{\gamma}$: velocidad de guiñada de vehículo.

3 DESCRIPCIÓN DE LOS MÉTODOS DE ESTIMACIÓN

3.1 Introducción

La estimación del ángulo de deslizamiento lateral del vehículo ha sido un gran desafío desde la introducción de los primeros sistemas a bordo que controlan la estabilidad del vehículo, como por ejemplo el control electrónico de estabilidad (ESC).

En la actualidad, los sistemas de control de vehículos como el control de la dirección de la rueda, el control directo del momento de guiñada, controles de tracción avanzados y los controles de estabilidad se utilizan en conjunto para mejorar el rendimiento del vehículo y sobre todo la estabilidad. Estos sistemas mejoran la capacidad de agarre en carretera, rendimiento en curvas, la capacidad de conducción y la estabilidad, ayudando al conductor especialmente durante las maniobras de emergencia o en situaciones críticas. Todos estos controles se basan en evaluaciones en tiempo real del estado del vehículo y del seguimiento del ángulo de deslizamiento lateral en particular, por lo que una estimación rápida y precisa de puede considerarse clave para la seguridad. [5]

El ángulo de deslizamiento lateral depende de los parámetros mencionados en el apartado 2.1.2, de los cuales existen sensores para poder medir estos parámetros como por ejemplo GPS para la velocidad longitudinal y lateral, giroscopios que proporcionan la velocidad angular del vehículo, acelerómetros miden la aceleración lineal y angular, magnetómetros proporciona ángulo de Euler, potenciómetros rotatorios para conocer la dirección de giro, entre otros. Con estos parámetros existen una gran cantidad de estudios basados en redes neuronales, observadores lineales y no lineales, integrados con los sensores mencionados anteriormente.

3.2 Estimación basada en observadores

Este enfoque utiliza un modelo de referencia del vehículo para estimadores de estado, tanto para condiciones en estado estacionario como transitorio. La necesidad de un modelo implica que los resultados estén relacionados con la complejidad del modelo y con el conocimiento de sus parámetros. Para estas aplicaciones se usan modelos lineales y no lineales del vehículo. Los observadores que existen son el observador Luenberger, el observador de modo deslizante y el filtro de Kalman siendo este último el más utilizado.

En [6] desarrolla un observador para estimar el ángulo de deslizamiento lateral tomando en cuenta las no linealidades del sistema, tanto en el diseño como en el análisis teórico. Los objetivos de este diseño incluyen una reducción de la complejidad computacional en comparación con la EKF. El diseño es basado en una configuración de sensor estándar, y está sujeto a pruebas exhaustivas en condiciones realistas.

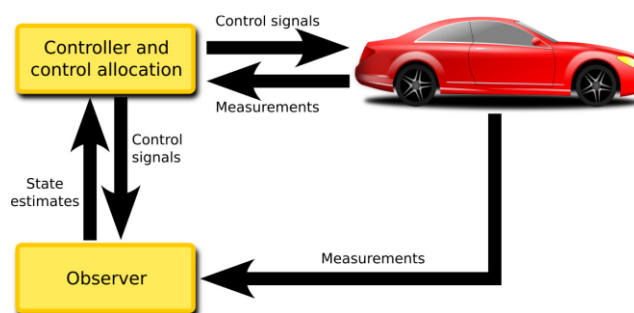


Figura 3-1: Esquema de estudio [6]

En [7] realiza una comparación entre el filtro de Kalman EKF y UKF, en la figura 3-1 se muestra el modelo de este estudio. Para la estimación utiliza la velocidad de guiñada y balanceo medido por giroscopios, aceleraciones laterales y longitudinales medidas por acelerómetros, deflexiones de suspensión medidos con sensores específicos, ángulo de dirección medido por un sensor óptico y la velocidad de rotación para cada rueda dada por sensores magnéticos. El primer bloque tiene como objetivo proporcionar la masa del vehículo, transferencia de carga lateral, las fuerzas de los neumáticos y la aceleración lateral corregida.

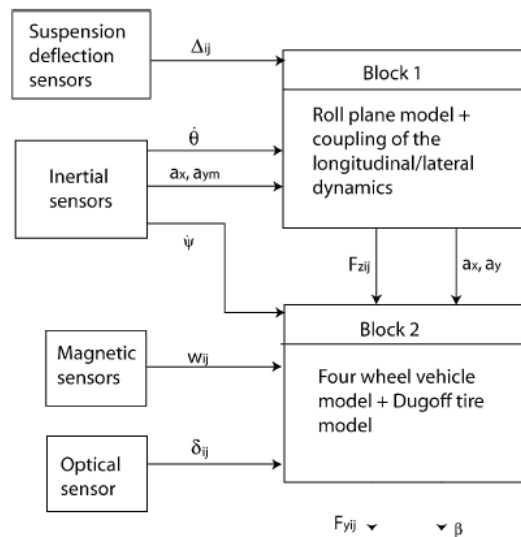


Figura 3-2: Esquema de estudio [7]

En [8] se propone un método de estimación del ángulo de deslizamiento lateral que considera la variación severa de la velocidad longitudinal durante un corto periodo de tiempo el cual un vehículo puede perder estabilidad debido al deslizamiento o atascamiento. Se utiliza un filtro de Kalman extendido (EKF) basado en un modelo cinemático.

En [9] utiliza mediciones de un GPS y un sistema de navegación inercial (INS) que se integra con un filtro de Kalman para proporcionar estimaciones de los parámetros del vehículo que se utilizan para calcular el ángulo de deslizamiento lateral.

En [10] utiliza un método de estimación de fusión del ángulo de deslizamiento lateral basado en un filtro Kalman y una relación de acoplamiento de velocidad de la rueda.

En [11] se utiliza un observador Luenberger para estimar los estados dinámicos del vehículo.

En [12] realiza la estimación del ángulo de deslizamiento lateral del vehículo con un filtro de Kalman EKF con un modelo dinámico del vehículo. Para comprobar los resultados utiliza CarSim y Simulink.

En [13] realiza una estimación mediante el algoritmo de filtro de Kalman EKF considerando características no lineales del neumático y se verificó mediante CarSim y Simulink, además de pruebas con un vehículo real.

En [14] se utiliza un observador no lineal para la estimación de la velocidad longitudinal, velocidad lateral y velocidad de guiñada de un vehículo. El estimador se basa en el modelo de fuerzas de un neumático.

En [15] presenta un método que estima el ángulo de deslizamiento lateral del vehículo y un coeficiente de fricción entre la llanta y la carretera mediante la combinación de medidas de un magnetómetro, un sistema de posicionamiento global (GPS) y una unidad de medición inercial (IMU). El algoritmo de estimación se basa en una estructura en cascada que consta de un marco de fusión de sensores basado en filtros de Kalman.

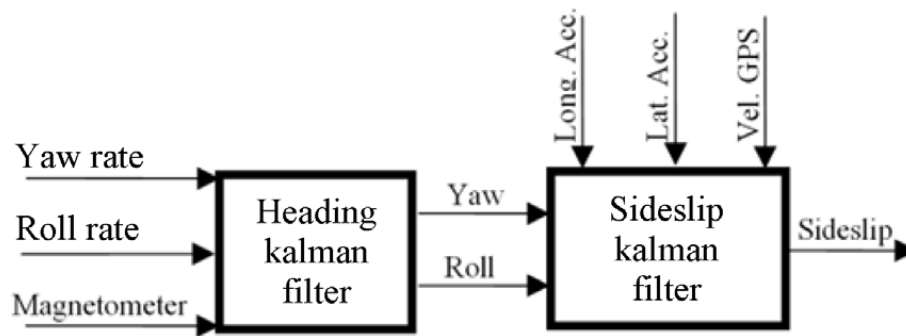


Figura 3-3: Esquema de estudio [15]

3.3 Redes Neuronales

Las redes neuronales se consideran herramientas eficaces para el modelado de sistemas, ya que son adecuadas para modelar sistemas complejos utilizando su capacidad para identificar relaciones entre datos de entrada y salida. También ofrecen ventajas decisivas como el aprendizaje adaptativo, la tolerancia a fallos y la generalización. En los últimos años, el desarrollo de computadoras de alta velocidad alentó a la aplicación de redes neuronales.

Con una red neuronal, el vehículo puede considerarse una caja negra y solo se necesita un conjunto convencional de sensores para entrenar y alimentar a la red. Las entradas de la caja negra son la velocidad o ángulo de guiñada, la aceleración lateral, el ángulo de dirección y la velocidad del vehículo, mientras que el ángulo de deslizamiento lateral es la salida.

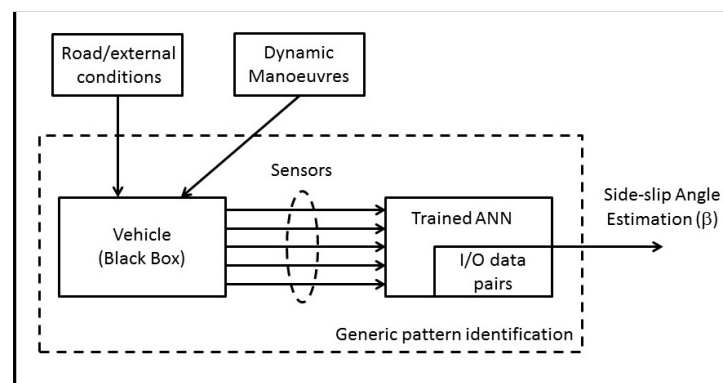


Figura 3-4: Esquema de alto nivel del proceso de estimación de redes neuronales artificiales (ANN)

En [5] se da un método fiable y eficaz para estimar el ángulo de deslizamiento lateral del vehículo mediante una red artificial. En esta aplicación la red es simple para poder trabajar en tiempo real dentro de aplicaciones integradas como sistemas de seguridad activa.

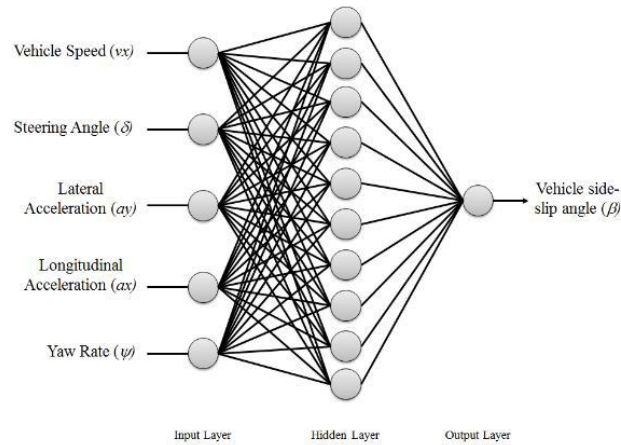


Figura 3-5: Red neuronal de [5].

En [16] se utiliza una red neuronal para estimar el ángulo de deslizamiento lateral del vehículo mediante una red neuronal, tomando como entradas la aceleración lateral y la aceleración del ángulo de guiñada. Sin embargo, la red neuronal BP tiene desventajas de velocidad de convergencia lenta y valor mínimo local. Este método utiliza una función de aprendizaje para mejorar la velocidad de estimación y tener un aprendizaje mas eficiente.

En [17] propone un observador basado en ANFIS (sistema de interferencia neuro-difuso adaptativo) combinado con Filtros de Kalman para estimar el ángulo de deslizamiento lateral, que a su vez se utiliza para controlar la dinámica del vehículo y mejorar su comportamiento.

3.4 Patentes de fabricantes

En [18], una patente de Toyota, se realiza una estimación del ángulo de deslizamiento para un vehículo mediante un dispositivo de formación de imágenes para capturar una imagen de al menos una de las partes delantera y trasera del vehículo y una unidad de control. El dispositivo de imágenes es una cámara CCD que incluye una lente y un sensor de imágenes. La unidad de control está configurada para calcular un ángulo de deslizamiento del vehículo basado en una relación de distancia horizontal entre un centro de imagen y el punto de fuga a una distancia entre el centro de la lente de la cámara CCD y un sensor de imagen.

En [19], una patente de Ford, se usa un método y sistema para determinar un ángulo de deslizamiento lateral para un vehículo automotor, que incluye varios sensores tales como un sensor de velocidad de guiñada, un sensor de velocidad, un sensor de aceleración lateral, un sensor de velocidad, un sensor de ángulo de dirección y un sensor de aceleración longitudinal. Cada uno de los sensores está acoplado a un controlador que determina una velocidad de ángulo de deslizamiento lateral en respuesta a las señales del sensor. La velocidad del ángulo de deslizamiento lateral se compensa debido a la gravedad y los cambios de actitud del vehículo. Además, la velocidad del ángulo de deslizamiento lateral se compensa debido a la no linealidad del ángulo de deslizamiento lateral. La velocidad del ángulo de deslizamiento lateral está integrada, preferiblemente con un filtro de integración anti-deriva (para determinar un ángulo de deslizamiento lateral integrado). También se determina un ángulo de deslizamiento lateral en estado estable en función de los sensores, como el sensor de velocidad de guiñada y el sensor de aceleración lateral. El ángulo de deslizamiento lateral en estado estacionario se filtra usando un filtro de recuperación en estado estacionario. El ángulo de deslizamiento lateral integrado y el ángulo de deslizamiento lateral en estado estable se combinan para formar una estimación del ángulo de deslizamiento lateral.

4 DISEÑO Y VALIDACIÓN DE LOS ESTIMADORES

4.1 Introducción

El ángulo de deslizamiento lateral (slip angle) es uno de los parámetros clave para el funcionamiento de los sistemas de estabilidad. En este capítulo se diseñarán algunos métodos para estimar el ángulo de deslizamiento lateral, para lo que se utiliza parámetros que normalmente son conocidos en los vehículos producidos en masas como son el ángulo de giro, velocidad longitudinal, y velocidad de guiñada.

4.2 Observador Luenberger

En la ecuación de la fórmula 4-3, se puede observar que V_x ingresa como un parámetro a la matriz A . Por otro lado los coeficientes de rigidez de las ruedas varían dependiendo la condición del suelo, tipo de neumático, la presión de estos, entre otros aspectos. Para este motivo es necesario que el observador sea robusto ante variaciones paramétricas del sistema y que permita compensar las dinámicas no modeladas.

El observador de Luenberger es un estimador de estado, un algoritmo que permite estimar el estado interno oculto, no medible, de un sistema dinámico lineal a partir de las mediciones de la entrada y la salida de dicho sistema.

En primer lugar, se discutirá la observabilidad del sistema. La matriz de observabilidad se construye a partir de la matriz A y C como Ob . Se puede probar usando Matlab que el rango de la matriz de observabilidad, en este caso, es 2 (rango completo) y, por lo tanto, el sistema es observable. La estructura del observador se presenta en la ecuación 4-4 donde se conserva la parte principal del sistema y se agrega un término de corrección $L(y - \hat{y})$ que hace que el sistema converja. La figura 4-2 presenta la estructura del observador donde la entrada del observador es la entrada u y la salida y del sistema. [11]

$$\begin{cases} \dot{\hat{x}} = A\hat{x} + Bu - L(y - \hat{y}) \\ y = C\hat{x} + Du \end{cases} \quad (4-1)$$

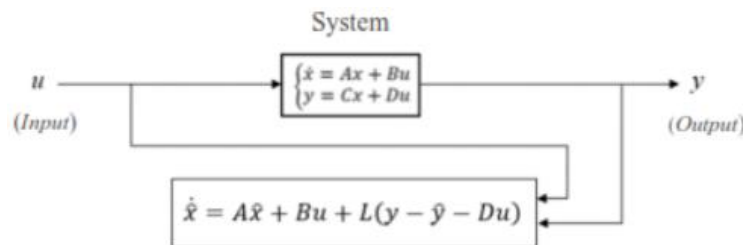


Figura 4-1: Estructura del Observador

El error dinámico del observador y del modelo se puede describir como se indica en la fórmula 4-2.

$$\begin{aligned}\dot{x} - \dot{\hat{x}} &= A(x - \hat{x}) - L(Cx - C\hat{x}) \\ &= (L - LC)(x - \hat{x})\end{aligned}\quad (4-2)$$

Para calcular el valor de L para estabilizar el sistema, se usa el método de ubicación de polos. La estructura final del observador se muestra en la figura 4-2. [20]

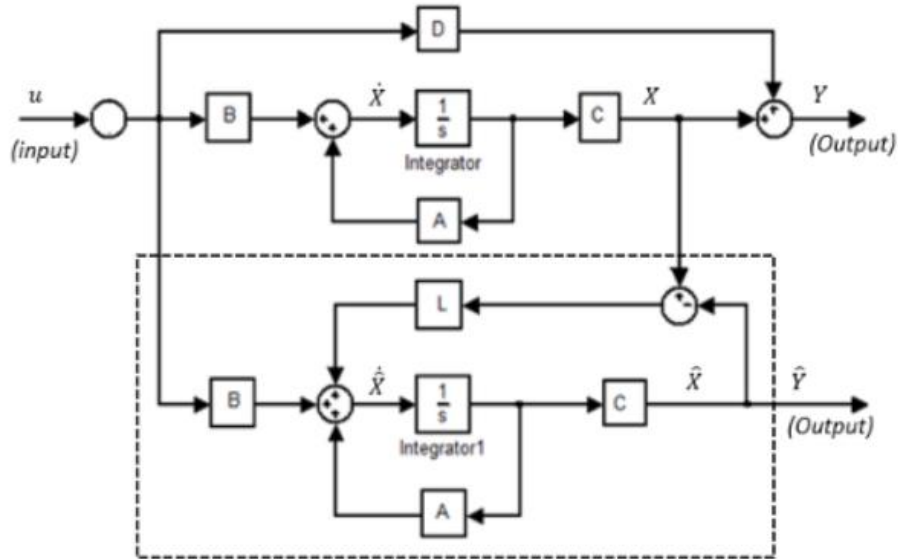


Figura 4-2: Esquema de estados del Observador Luenberger

4.3 Filtros de Kalman

El filtro de Kalman es una técnica bien conocida que sirve para poder estimar el estado oculto o no medible de un sistema dinámico. Aunque al principio fue introducido como solución a problemas de ingeniería eléctrica, el filtro de Kalman rápidamente fue utilizado en aplicaciones de navegación.

Consiste esencialmente en una serie de ecuaciones matemáticas que implementan un estimador tipo predictor – corrector que es óptimo en el sentido de que minimiza el error estimado de la covarianza, cuando algunas condiciones son dadas. Desde el momento de su introducción, el filtro de Kalman ha sido sujeto de investigación extensiva y aplicaciones, en particular en el área de los vehículos autónomos o la navegación asistida. Aunque no siempre se presentan todas las condiciones óptimas para el funcionamiento del filtro, éste se desempeña bien en la mayoría de situaciones.

El filtro de Kalman realiza una combinación estadística de las medidas y las predicciones basadas en el modelo del sistema, tratando ambas como una secuencia de unas estimaciones ruidosas o con cierto grado de incertidumbre.

Otra de las ventajas de esta técnica es que incorpora toda la información que pueda dársele, procesando todas las mediciones disponibles. Es por esto que el filtro de Kalman es el estimador más utilizado con la fusión de sensores de un sistema. Además, no requiere que todos los datos previos sean almacenados y reprocesados cada vez que se toma una nueva medida. [21]

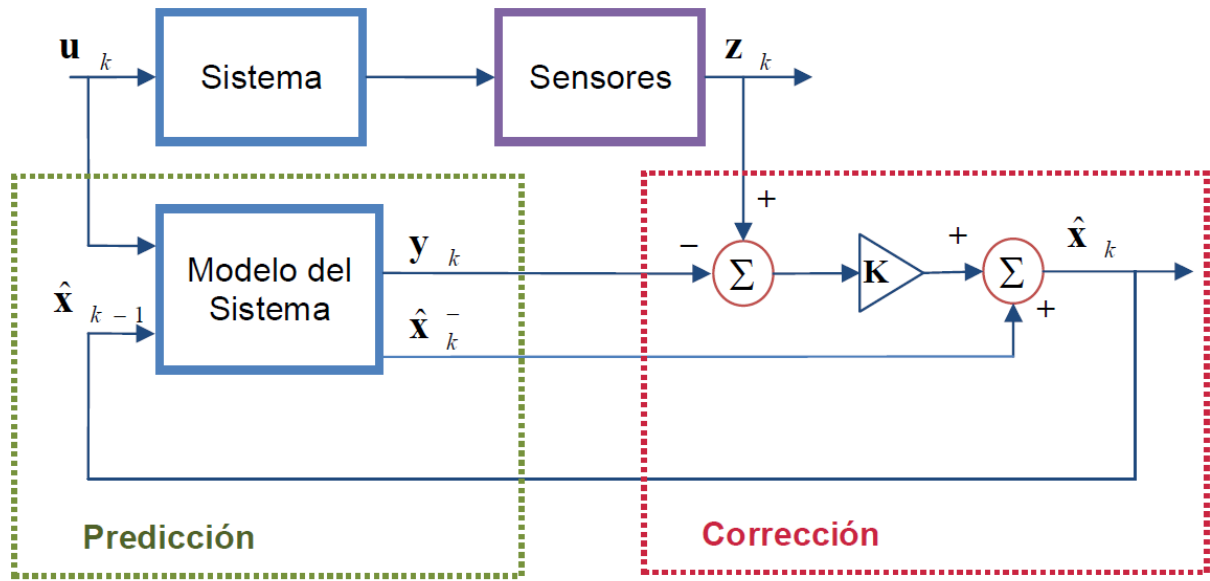


Figura 4-3: Etapas del Filtro de Kalman.

El sistema no lineal gobernado por las ecuaciones diferenciales estocásticas no lineales se puede escribir como:

$$\begin{aligned} x_{k+1} &= f_k(x_k, u) + w_k \\ y_k &= h_k(x_k, u) + v_k \end{aligned} \quad (4-3)$$

Donde:

- x_k representa el vector de estados $[\beta, \Upsilon]^T$
- u es la entrada $[\delta]$
- w_k el vector de ruido del proceso
- y_k el vector de medida
- v_k el vector de ruido de medición.

Se supone que w_k y v_k son blancos, de media cero y no correlacionados

$$\begin{aligned} w_k &\sim N(0, Q_k) \\ v_k &\sim N(0, R_k) \end{aligned} \quad (4-4)$$

Donde:

- Q y R son las matrices de covarianza que describen las propiedades de segundo orden del ruido del estado y medición.

Utilizando las matrices discretizadas del modelo lineal del vehículo el sistema se representa:

$$\begin{aligned}x_{k+1} &= Ax_k + Bu + w_k \\ y_k &= Cx_k + v_k\end{aligned}\tag{4-5}$$

El filtro de Kalman se resume en las siguientes ecuaciones:

1. La predicción de estado:

$$\tilde{x}_{k|k-1} = A\tilde{x}_{k-1|k-1} + Bu_k\tag{4-6}$$

2. La covarianza de error predictiva se calcula como:

$$P_{k|k-1} = AP_{k-1|k-1}A^T + Q\tag{4-7}$$

3. La ganancia de Kalman se calcula:

$$K_x = AP_{k|k-1} + H^T[HP_{k|k-1}H^T + R]^{-1}\tag{4-8}$$

4. La estimación del estado se actualiza con la medición y_k :

$$\tilde{x}_{k|k} = \tilde{x}_{k|k-1} + H_k[y_k - H\tilde{x}_{k|k-1}]\tag{4-9}$$

5. La covarianza de error se actualiza:

$$P_{k|k} = [I - K_k H]P_{k|k-1}\tag{4-10}$$

4.4 Redes Neuronales

4.4.1 Introducción

Las redes neuronales artificiales (RNA) se basan en el funcionamiento del sistema neuronal del cuerpo humano. En el que encontramos tres elementos principales: los órganos receptores que recogen la información del exterior; el sistema nervioso que transmite la información, la analiza y en parte la almacena, y envía la información elaborada y, los órganos actuadores que reciben la información de parte del sistema nervioso y la convierte en una acción.

La unidad fundamental del sistema es la neurona, estas se unen entre ellas formando redes. Se componen de un cuerpo o núcleo, del axón, que es una ramificación de salida de la neurona, y de un gran número de ramificaciones de entrada llamadas dendritas. El funcionamiento es que las señales de entrada llegan a la neurona a través de la sinapsis, que es la zona de contacto entre neuronas (u otro tipo de células, como las receptoras), que recoge la información electro-química procedente de las células adyacentes que están conectadas a la neurona; esta información llega al núcleo de la neurona a través de las dendritas, que la procesa hasta generar una respuesta, la cual es posteriormente propagada por el axón.

La sinapsis está compuesta de un espacio líquido donde existe una cierta concentración de iones, que tiene unas determinadas características eléctricas que permiten inhibir o potenciar la señal a conveniencia.

El sistema neuronal es un conjunto de neuronas conectadas entre sí, que reciben, elaboran y transmiten información a otras neuronas, y que dicha información se ve potenciada o inhibida en la siguiente neurona a conveniencia, gracias a las propiedades del espacio intersináptico, esta propiedad de poder alterar el peso de cada información en la red neuronal nos otorga en cierta medida la capacidad de aprender. [22]

4.4.2 Redes neuronales artificiales

Las redes neuronales artificiales tratan de emular las características y propiedades de las redes neuronales biológicas. Consisten de una serie de unidades denominadas neuronas, conectadas entre sí.

Las neuronas se conectan entre sí según una determinada arquitectura. Cada conexión tiene un determinado peso que pondera cada entrada a la neurona, de esta manera la entrada de cada neurona es la suma de las salidas de las neuronas conectadas a ella, multiplicadas por el peso de la respectiva conexión como se ve en la figura 4-5.

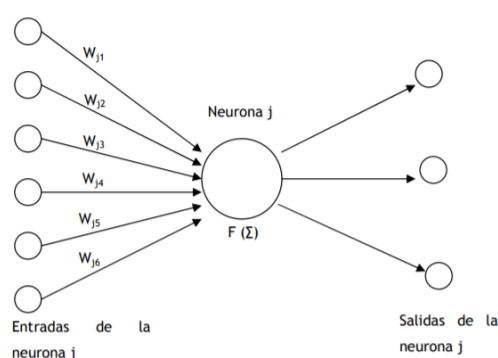


Figura 4-4: Esquema de funcionamiento de una neurona

En este modelo, la neurona j recibe una serie de entradas $x_1, x_2, \dots, x_n$. Cada señal se multiplica por el peso asociado a su conexión, $w_1, w_2, \dots, w_n$. Luego, se suman estas entradas ponderadas y se les aplica la función de activación $F(\cdot)$ para generar la señal de salida de la neurona j . Los valores de los pesos son ajustados durante la

fase de aprendizaje.

En cada arquitectura las neuronas se agrupan en distintas capas: una capa de entrada, otra de salida, y en el caso de existir, una o varias capas ocultas.

Las entradas a la red son introducidas en las neuronas de la capa de entrada, que genera una salida tal cual o las escala para que las señales se encuentren en un determinado rango. Estas entradas son propagadas a las neuronas de la siguiente capa. Según la figura 4-5 cada neurona j de la segunda capa generará una salida de valor:

$$S_{2j} = F_{2j}(x_1 W_{1j}) \quad (4-11)$$

Donde:

- x_1 es el vector de entradas de la capa 1
- W_{1j} es el vector de pesos correspondientes a las conexiones que van de todas las neuronas de la primera capa a la neurona j de la segunda capa.
- La función F_{2j} es la función de activación de la neurona j de la segunda capa.

De la misma manera con todas las neuronas de la segunda capa. Estas salidas son propagadas a las neuronas de la capa de salida, las que generan las salidas de la red. Cada neurona i de la capa de salida generará una salida de valor:

$$S_{3i} = F_{3i}(W_{2i} S_2) \quad (4-12)$$

Donde:

- W_{2i} es el vector de pesos correspondientes a las conexiones que van de las neuronas de la segunda capa a la neurona i de la capa de salida.
- S_2 es el vector de salidas de las neuronas de la capa dos, que a su vez son entradas de las neuronas de la capa de salida.

Las redes neuronales (RNA) son sistemas de aprendizaje basadas en datos que son utilizados como patrones. Es por esto que la capacidad de una red para resolver un problema está vinculada a los patrones utilizados durante su fase de aprendizaje.

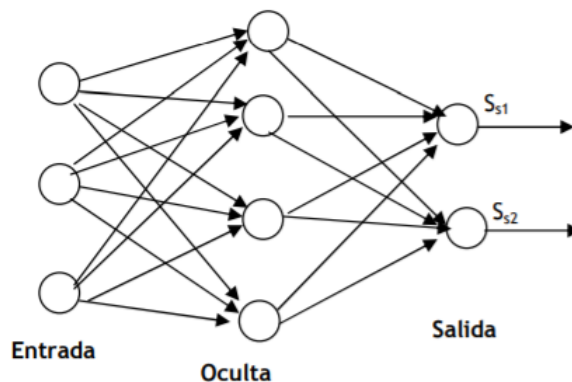


Figura 4-5: Esquema de la arquitectura de una red

El aprendizaje de una red neuronal consiste en encontrar los valores precisos de los pesos de sus conexiones para que pueda resolver un determinado problema. El proceso general consiste en ir introduciendo una serie de datos patrón y ajustar los pesos siguiendo un determinado criterio. Los criterios que se van a utilizar en este proyecto se fundamentan en el error cometido por la red, lo que nos obliga a conocer la salida que se debería obtener para cada uno de ellos. Es lo que se conoce como entrenamiento supervisado. De esta manera, primero se introducen los patrones, se reajustan los pesos, posteriormente se comprueba si se ha cumplido un determinado criterio de convergencia, de no ser así se repite todo el proceso. [22]

4.4.3 Método usado para la estimación

El enfoque utilizado para esta estimación es el mismo adoptado por [23] que utiliza un algoritmo de retropropagación y el índice de rendimiento del error cuadrático medio (MSE).

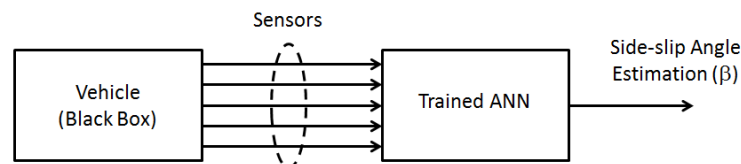


Figura 4-6: Esquema de alto nivel

El desempeño de la red neuronal se determina por el MSE que es:

$$MSE = \frac{1}{N} \sum_{i=1}^N (y_i - y_k)^2 \quad (4-13)$$

Donde:

- y_i es el valor predictivo
- y_k es el valor objetivo del i-ésimo patrón
- N es el número de patrón.

El conjunto de datos necesarios para entrenar y probar la RNA debe estar compuesto por patrones de datos de entrada junto con los objetivos correspondientes. Además, debe dividirse en dos conjuntos diferentes, el conjunto de datos que se utiliza para entrenar la red y el conjunto de datos que se utiliza para probar la capacidad de la red.

Se supone que este estimador es confiable y liviano desde el punto de vista computacional, por lo que puede usarse en aplicaciones integradas en tiempo real. El número de neuronas de la capa oculta conduce a satisfacer los propósitos de precisión y simplicidad. Se ha elegido utilizar un enfoque iterativo de "prueba y error". El diseño de red resultante capa de entrada-oculta-salida se ha creado utilizando la librería torch.nn con Phytton.

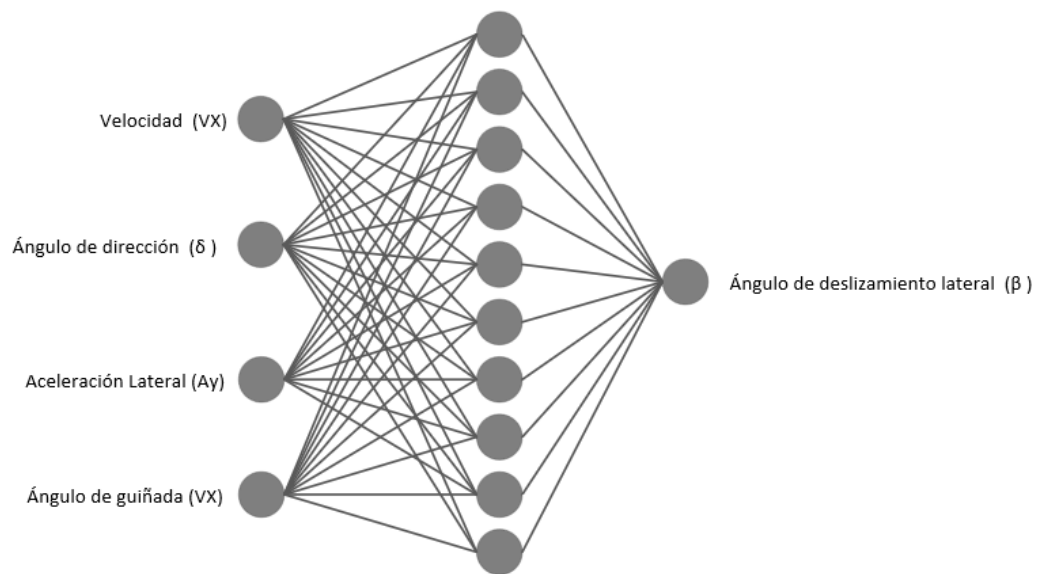


Figura 4-7: Estructura de red neuronal

5 MODELOS DE ESTIMACIÓN DEL ANGULO DE DESLIZAMIENTO LATERAL

5.1 Introducción

Este proyecto pretende modelar algunos sistemas para estimar el ángulo de deslizamiento lateral de un vehículo. Este se implementará en el software Simulink y la red neuronal en Phyton, y se someterá a estudio un modelo de vehículo seleccionado en el software de CarSim, sobre el que se realizarán una serie de pruebas y se analizarán los resultados obtenidos.

Dado que este sistema, para la estimación del ángulo β , en un primer lugar, se seleccionará un modelo de vehículo en el software CarSim como sujeto de estudio, y se obtendrán los parámetros necesarios para el diseño de los modelos a partir de sus características técnicas.

5.2 Características del vehículo

Para el estudio de los modelos para la estimación del ángulo de deslizamiento lateral, se escogió un vehículo en el software de CarSim. En este caso se escogió un deportivo clase B como se ve en la figura 5-1. Los parámetros del modelo seleccionado son los siguientes:

- Coeficientes de rigidez lateral de los neumáticos, se considera igual para el eje trasero como el delantero.
- Distancia del eje delantero y trasero al centro de gravedad del vehículo.
- Masa total del vehículo, que será la suma de la masa suspendida y no suspendida.
- Momento de inercia alrededor del eje vertical sobre el centro de gravedad.

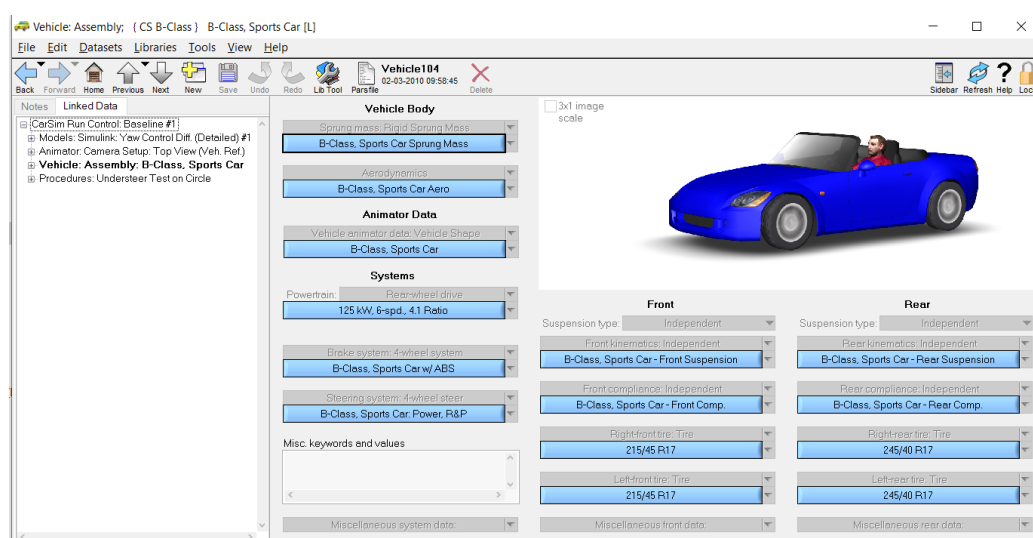


Figura 5-1: Vehículo seleccionado

De la figura 6-2 se obtiene que la distancia entre ejes es de 2800 mm, y la distancia del centro de gravedad al eje delantero es de 1165 mm, por lo que se extraen finalmente los valores de l_f y l_r :

$$l_f = 1165 \text{ mm} \quad (5-1)$$

$$l_r = 1635 \text{ mm}$$

También se extrae el valor del momento de inercia alrededor del eje vertical denominado como Yaw Inertia (I_{zz}).

$$I_{zz} = 996 \text{ kg m}^2 \quad (5-2)$$

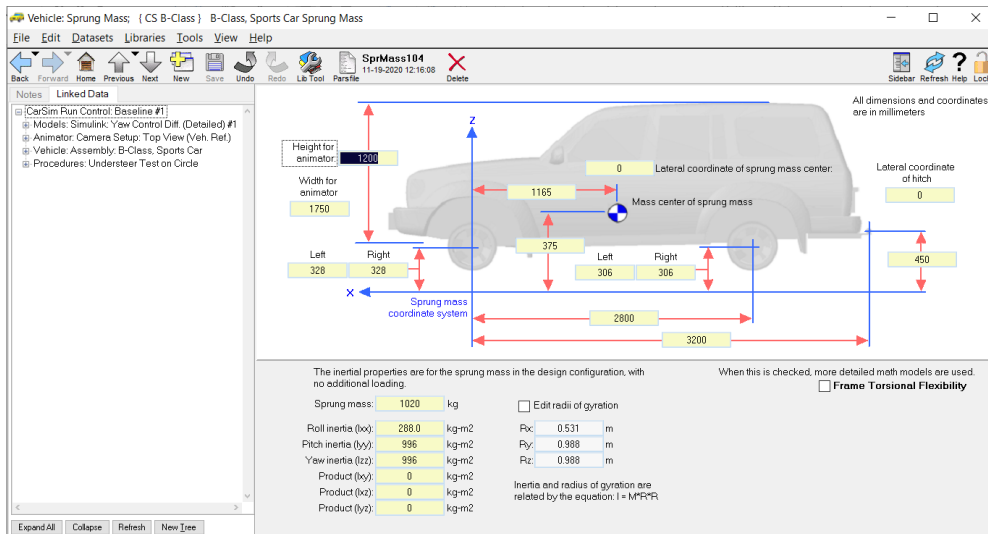


Figura 5-2: Vehículo seleccionado

Para el cálculo de la masa total del vehículo se tendrá en cuenta la masa suspendida, y las masas no suspendidas del eje delantero y el eje trasero.

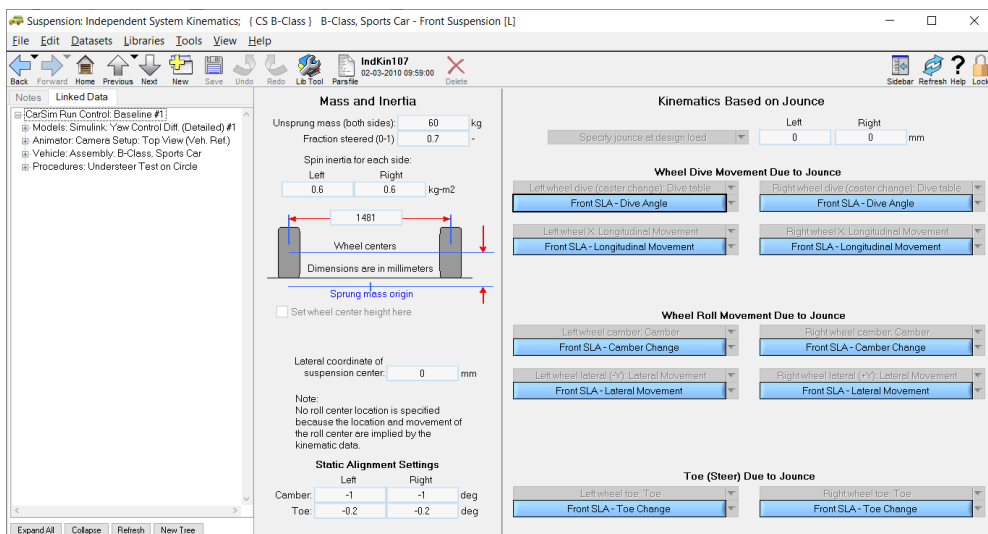


Figura 5-3: Información del eje delantero

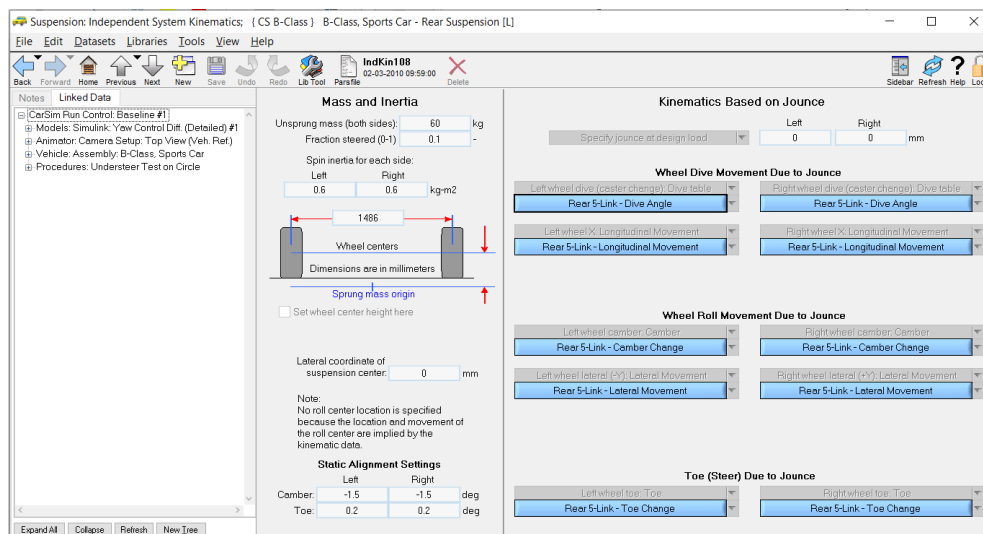


Figura 5-4: Información del eje trasero

De la figura 5-2 se obtiene el valor de la masa suspendida, y de las figuras 5-3 y 5-4 las masas no suspendidas del eje delantero y trasero respectivamente.

$$m = 1020 + 60 + 60 \text{ kg} = 1140 \text{ kg}$$

(5-3)

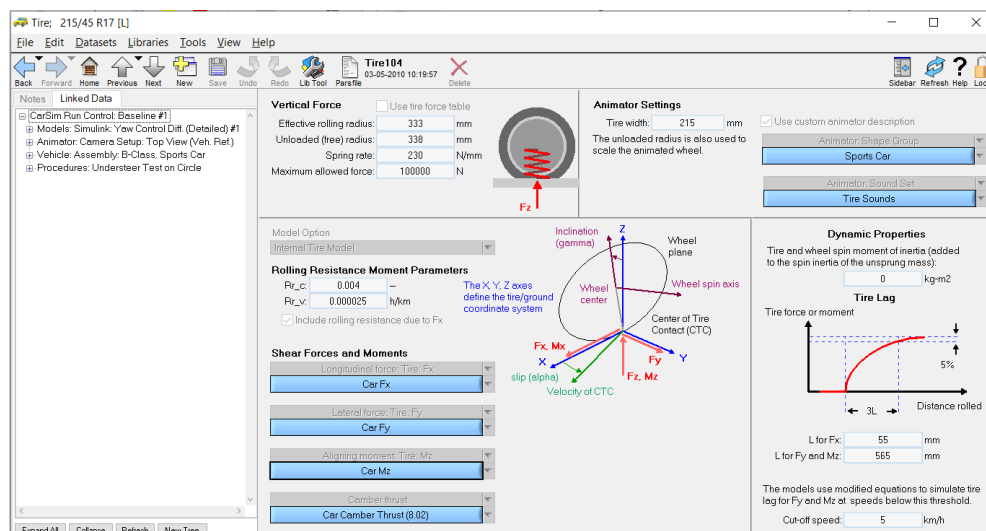


Figura 5-5: Información de los neumáticos

Para la obtención de los coeficientes de rigidez lateral se seleccionará la opción de CarSim referido a los neumáticos (figura 5-6), y después el desplegable para la fuerza lateral Car F_y .

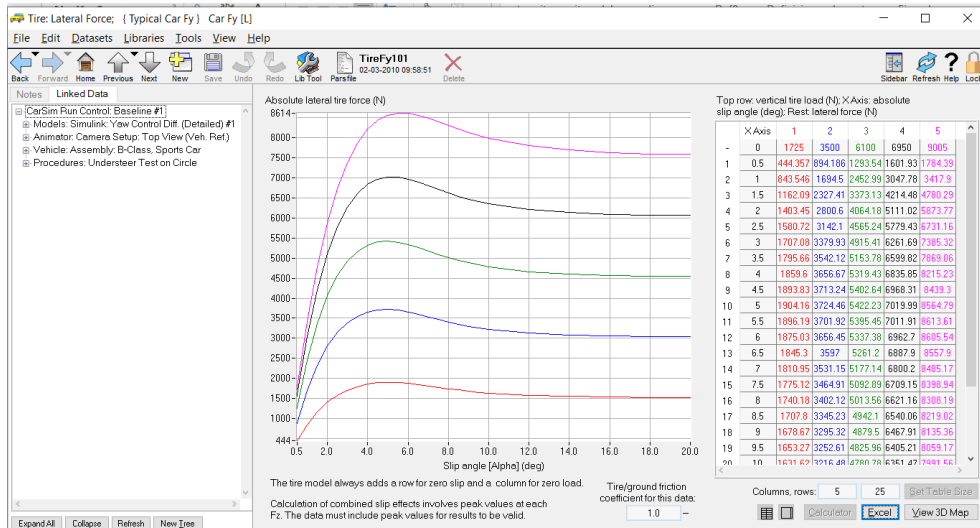


Figura 5-6: Información del eje trasero

La rigidez a deslizamiento lateral se entiende como la relación entre la fuerza lateral y el ángulo de deslizamiento lateral formado en el neumático. Esta relación presenta un comportamiento lineal para valores pequeños del ángulo, y se corresponde con la zona estable.

El software de CarSim proporciona las curvas de rigidez al deslizamiento lateral para diferentes condiciones de carga vertical sobre el neumático, y las representa en el siguiente diagrama en tres dimensiones.

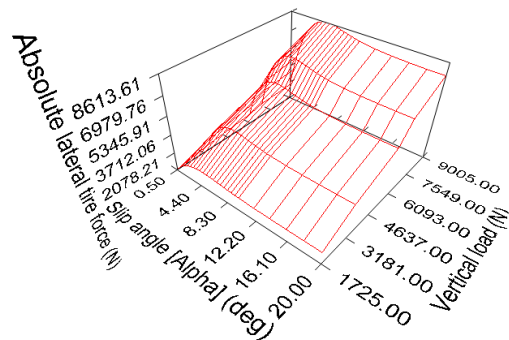


Figura 5-7: Información de los neumáticos

La rigidez a deslizamiento lateral se obtiene de la pendiente de la curva en la zona lineal, que corresponde a ángulos de deriva menores de $\alpha=5^\circ$. Se seleccionará una curva intermedia de la figura 5-7 y se calcula como el cociente entre la fuerza lateral F_y y el ángulo de deriva del neumático α , y se expresará en N/rad .

$$C_{af} = C_{ar} = \frac{5345.91 \text{ N}}{4 \text{ deg}} \cdot \frac{180 \text{ deg}}{\pi \text{ rad}} = 76575 \text{ N/rad} \quad (5-4)$$

Se presenta una tabla con los parámetros del modelo de CarSim que se usa para los modelos propuestos.

Tabla 5–1 Parámetros del vehículo Class-B de simulación

Parámetro	Nombre del parámetro	Valor [unidad]
m	Masa del vehículo	1140 [kg]
I_{zz}	Momento de inercia del vehículo alrededor del eje z	996 [kg m ²]
F_y	Fuerza lateral, sobre la componente y	5345.91 [N]
C_{af}	Rigidez a deslizamiento lateral en la rueda delantera	76575 [N/rad]
C_{ar}	Rigidez a deslizamiento lateral en la rueda trasera	76575 [N/rad]
l_f	Distancia entre el centro de gravedad y el eje delantero	1.165 [m]
l_r	Distancia entre el centro de gravedad y el eje trasero	1.635 [m]

5.3 Maniobras de las pruebas para simular

Para comprobar el comportamiento de los modelos de estimación del ángulo de deslizamiento lateral se escogieron maniobras en las que se presenten cambio de dirección en los que aparecen esfuerzos laterales sobre la dinámica del vehículo.

Las respuestas del sistema para la estimación del ángulo de deslizamiento lateral de los modelos propuestos se compararán respecto a la respuesta proporcionada por el software de simulación CarSim.

Las maniobras escogidas son doble cambio de carril, entrada sinusoidal y en círculo.

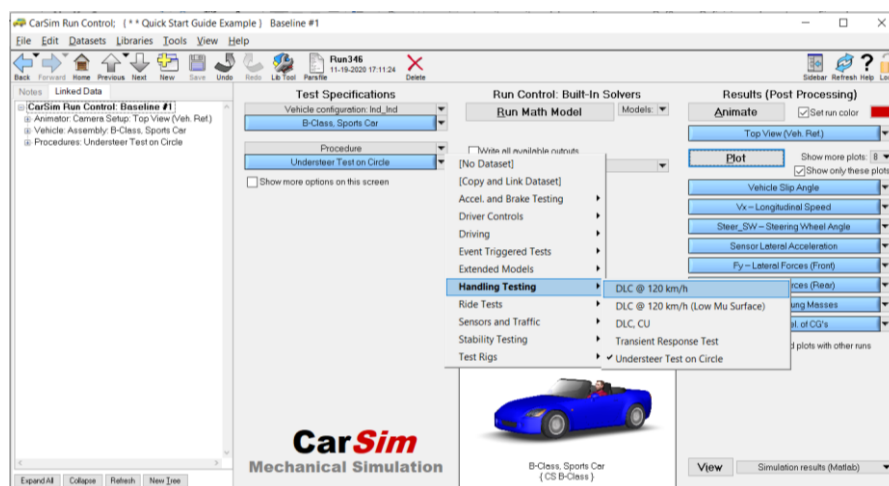


Figura 5-8: Maniobras en CarSim

5.3.1 Giro circular a 40 km/h

Durante esta maniobra el vehículo gira a 15° aproximadamente de manera constante de modo que permanece moviéndose en forma de una circunferencia a velocidad de 40 km/h.

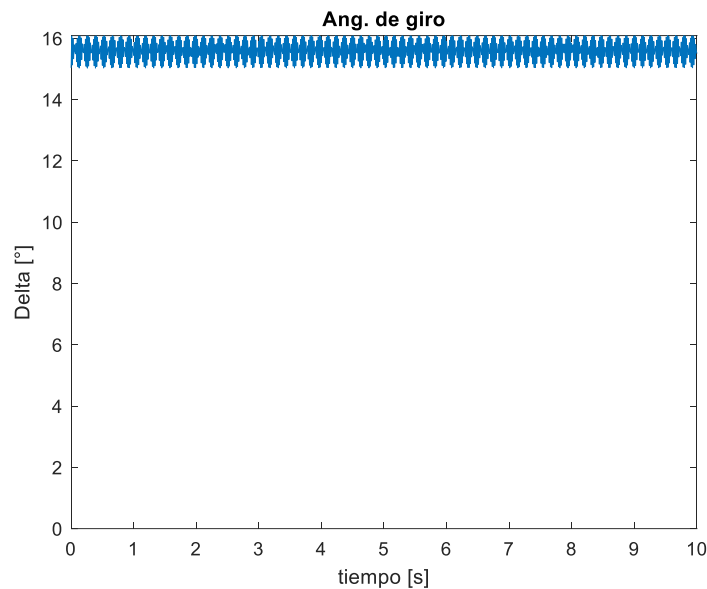


Figura 5-9: Ángulo de giro, simulación giro circular 40 km/h. Modelo lineal

5.3.2 Doble cambio de carril a 120 km/h

Durante esta maniobra el vehículo se encuentra realizando una trayectoria recta a una velocidad de 120 km/h, gira para cambiar de carril, luego gira en sentido contrario para volver al carril inicial.

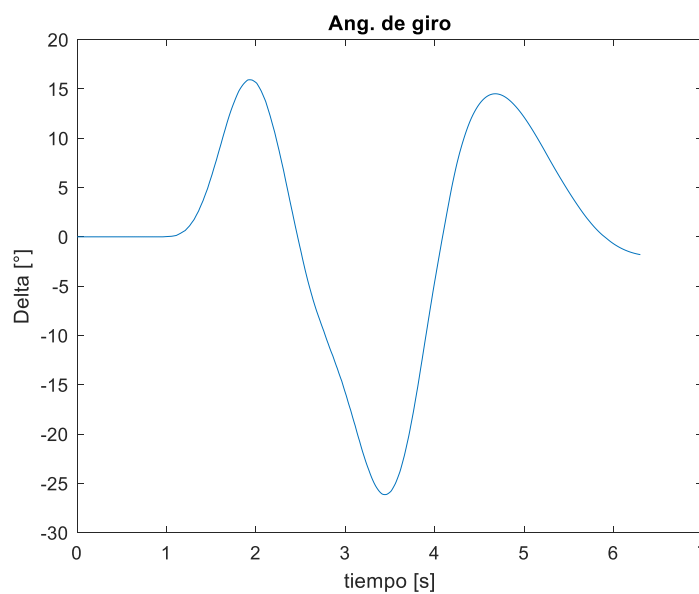


Figura 5-10: Ángulo de giro, simulación doble cambio de carril a 120 km/h. Modelo lineal.

5.3.3 Movimiento sinusoidal a 50 km/h

Durante esta maniobra el vehículo se encuentra realizando giros con tal que forma una trayectoria sinusoidal uniforme.

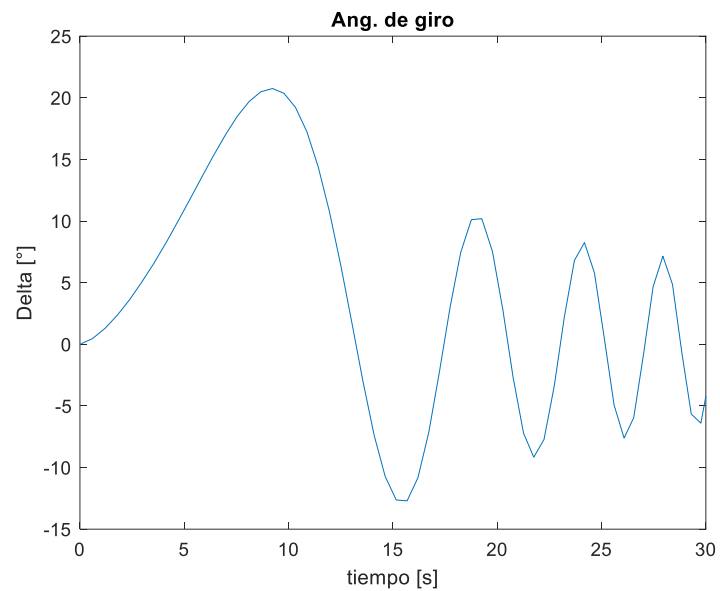


Figura 5-11: Ángulo de giro, movimiento sinusoidal. Modelo lineal

5.4 Modelo Lineal

Para construir el modelo lineal se utilizan las matrices de la fórmula 5-3 del modelo lineal del vehículo en el que la entrada es el ángulo de giro del volante (δ), y como salidas se tiene el ángulo de deslizamiento lateral (β) y el ángulo de guiñada (γ).

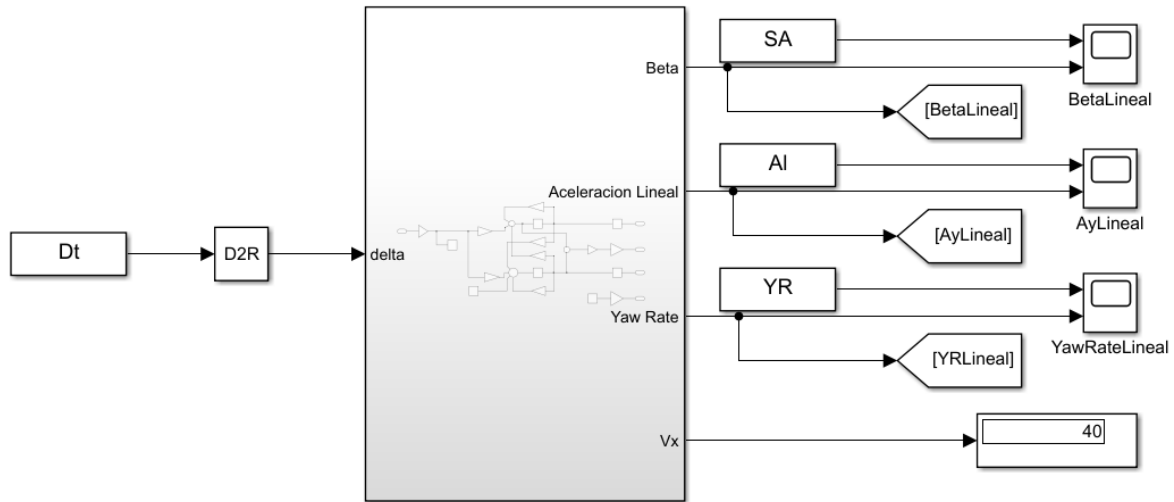


Figura 5-12: Sistema lineal Simulink

También mediante las ecuaciones de la fórmula 5-5 se puede obtener la aceleración lateral (a_y).

$$a_y = \dot{V}_y + V_x \gamma \quad (5-5)$$

$$\dot{V}_y = \dot{\beta} V_x$$

Entonces:

$$a_y = (\dot{\beta} + \gamma) V_x$$

El modelo del sistema lineal queda como se muestra en la figura 2-6.

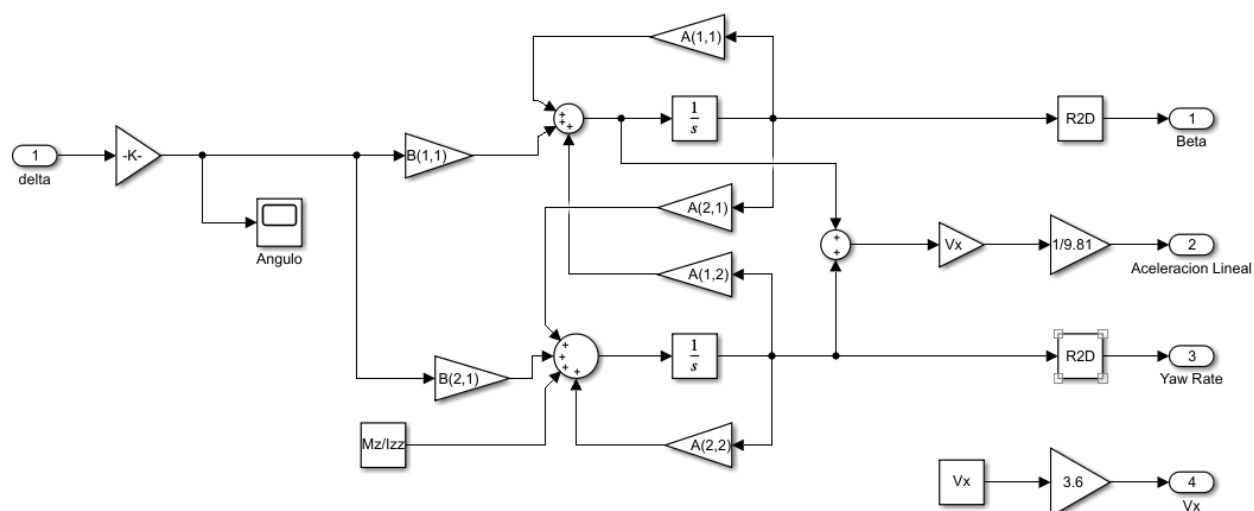


Figura 5-13: Bloque sistema lineal

5.4.1 Simulaciones

5.4.1.1 Giro circular a 40 km/h

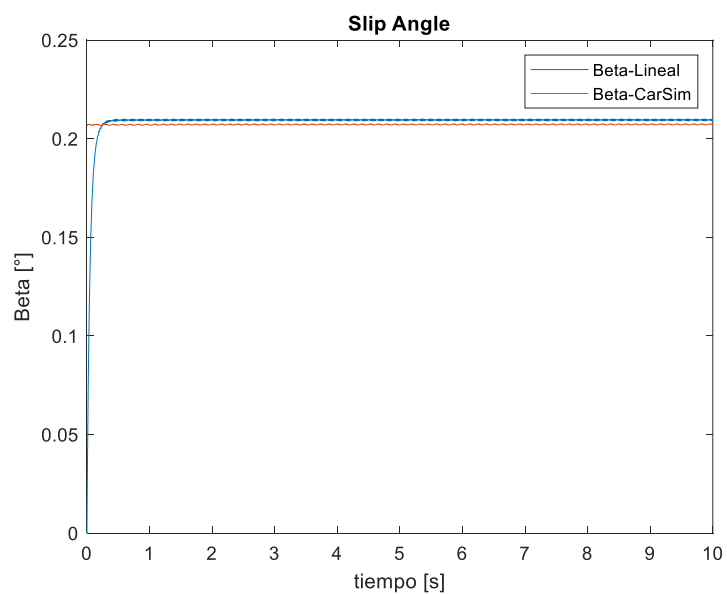


Figura 5-14: Ángulo de deslizamiento lateral, simulación giro circular 40 km/h. Modelo lineal

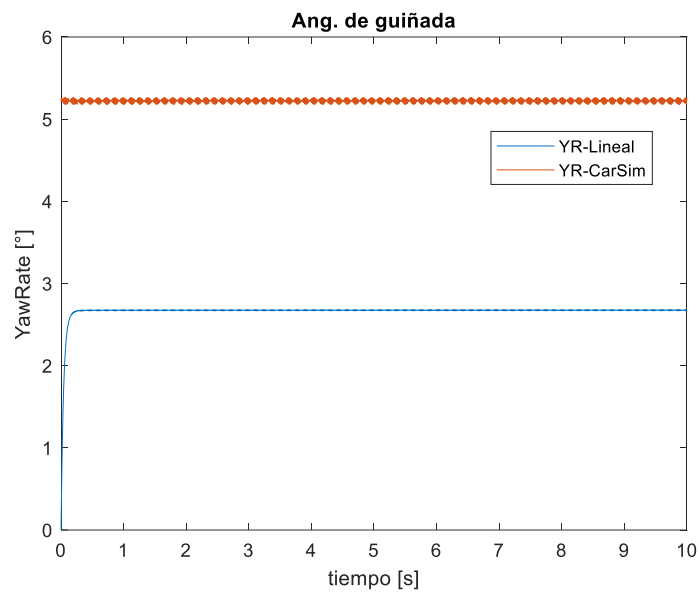


Figura 5-15: Ángulo de guiñada, simulación giro circular 40 km/h. Modelo lineal

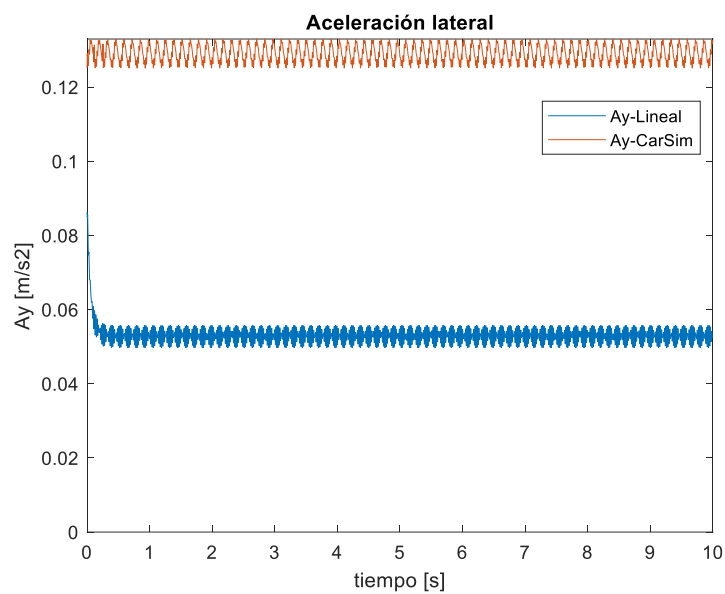


Figura 5-16: Aceleración lateral, simulación giro circular 40 km/h. Modelo lineal

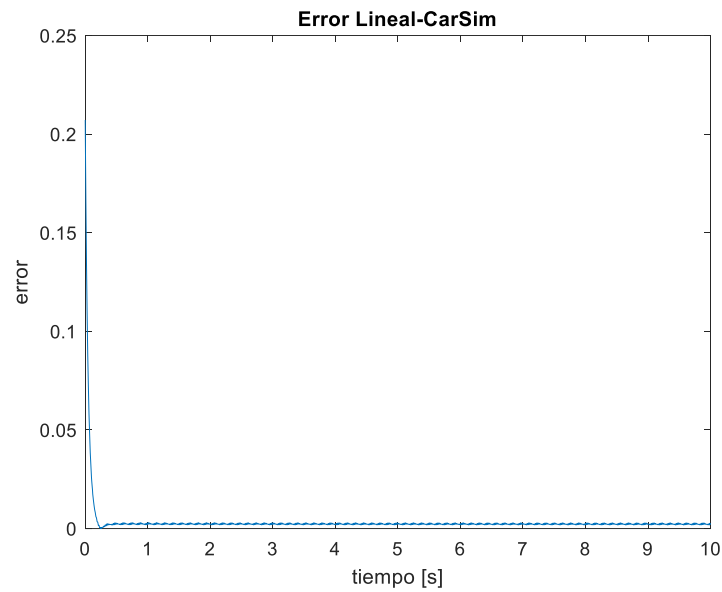


Figura 5-17: Error β Carsim respecto β de la simulación simulación giro circular 40 km/h.
Modelo lineal

Como se puede ver en la figura 5-17 el error entre β Carsim respecto β de la simulación simulación es mínimo.

5.4.1.2 Doble cambio de carril a 120 km/h

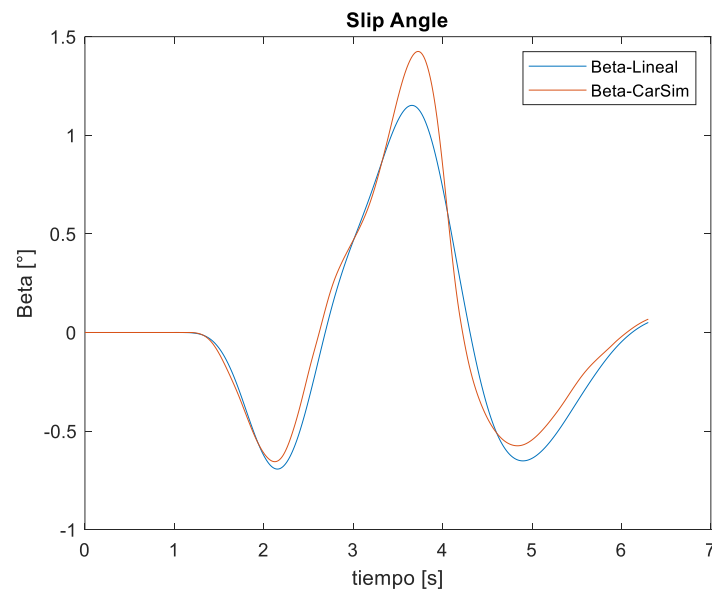


Figura 5-18: Ángulo de desplazamiento lateral, simulación doble cambio de carril a 120 km/h. Modelo lineal.

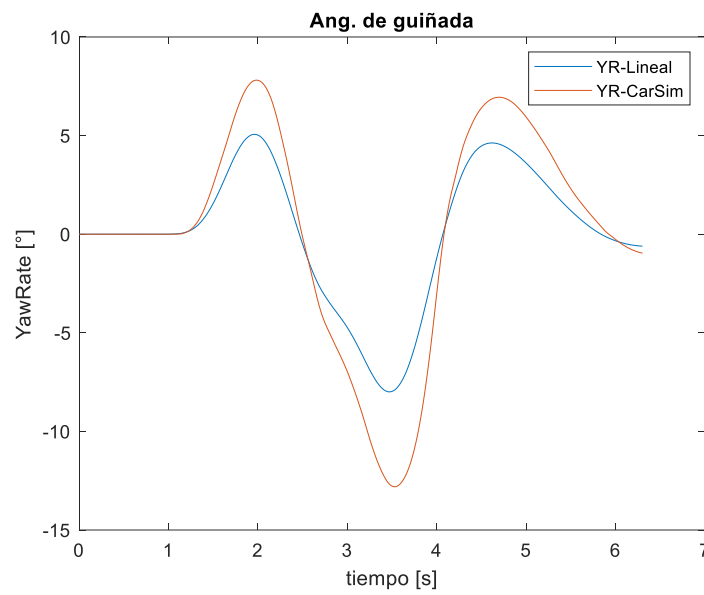


Figura 5-19: Ángulo de guiñada, simulación doble cambio de carril a 120 km/h. Modelo lineal.

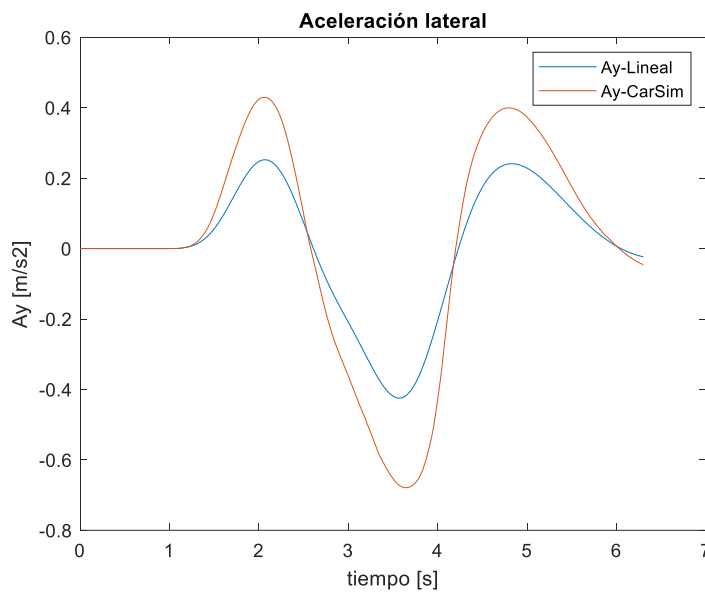


Figura 5-20: Aceleración lateral, simulación doble cambio de carril a 120 km/h. Modelo lineal.

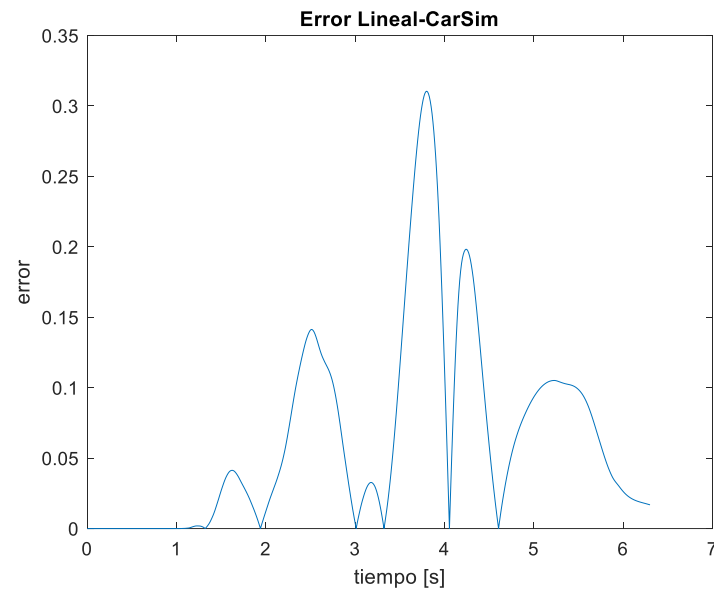


Figura 5-21: Error β Carsim respecto β de la simulación doble cambio de carril a 120 km/h.
Modelo lineal

El error β Carsim respecto β de la simulación doble cambio de carril es aceptable, se puede ver en la figura 5-18 el ángulo β estimado es aproximado al real.

5.4.1.3 Movimiento sinusoidal a 50 km/h

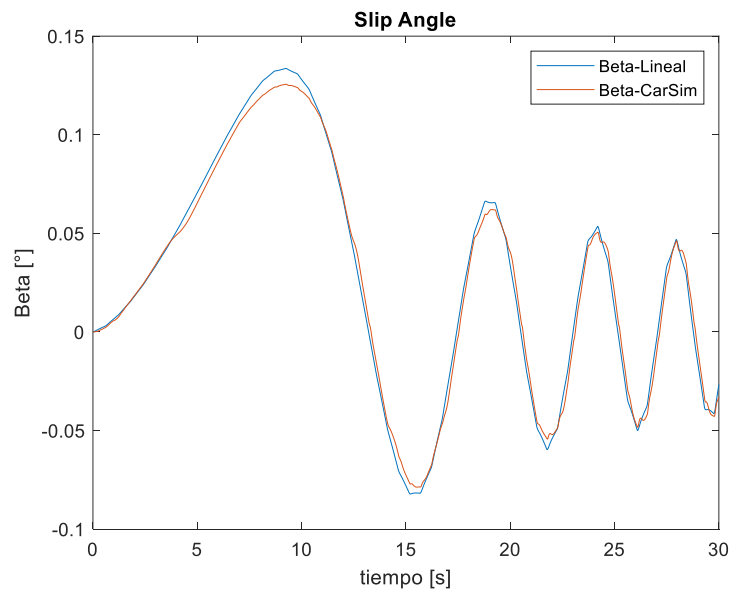


Figura 5-22: Ángulo de desplazamiento lateral, movimiento sinusoidal. Modelo lineal

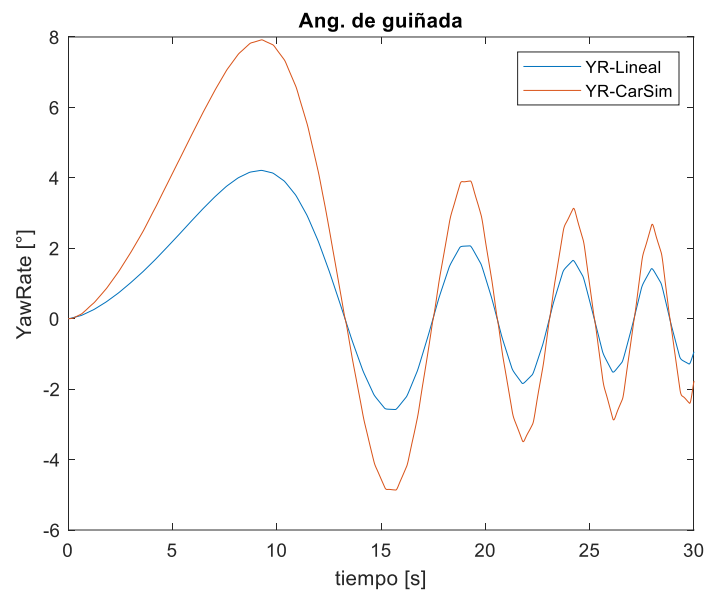


Figura 5-23: Ángulo de guiñada, movimiento sinusoidal. Modelo lineal

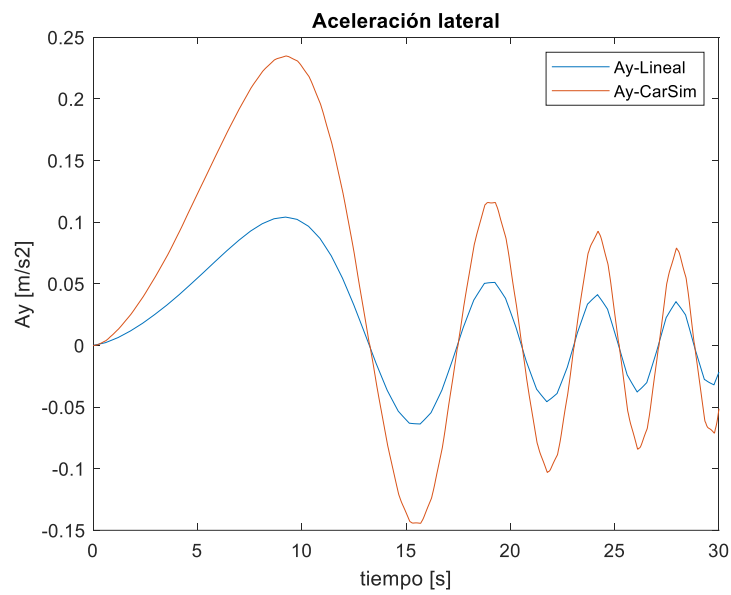


Figura 5-24: Aceleración lateral, movimiento sinusoidal. Modelo lineal

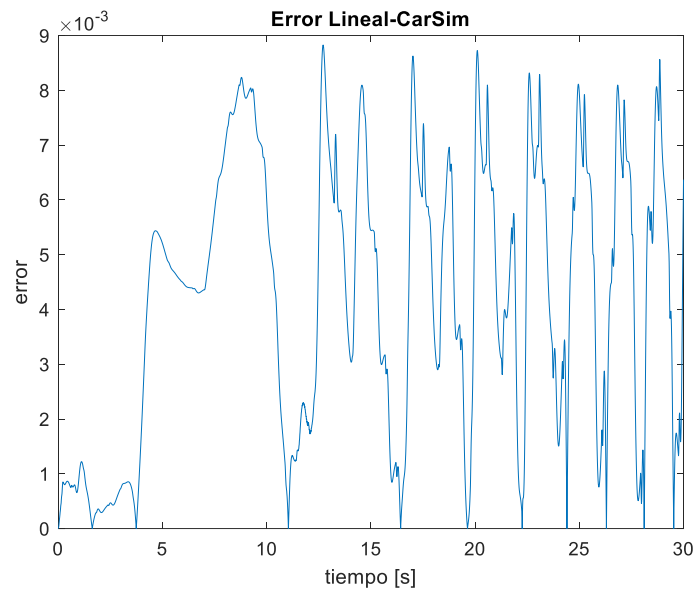


Figura 5-25: Error β Carsim respecto β de la simulación movimiento sinusoidal. Modelo lineal.

Como se puede observar en las respuestas del modelo lineal a la estimación del ángulo de deslizamiento lateral respecto al de CarSim que realiza una buena estimación, no tiene mucho error. Como se aprecia en la figura 5-22 el seguimiento de la β estimada es muy aproximada a la real.

5.5 Modelo basado en Observador Luenberger

Para construir el sistema con el observador Luenberger se utilizan las matrices de la fórmula 5-3 del modelo lineal del vehículo, en el que las entradas son el ángulo de giro del volante (δ) y el ángulo de guiñada (Y), que representan las señales que van desde los sensores del vehículo. Y como salidas se tiene el ángulo de deslizamiento lateral (β) y el ángulo de guiñada (Y) de seguimiento. La matriz L se calcula mediante el método de ubicación de polos, con el comando `lqry` se encuentra la matriz de ganancia óptima para cada caso dependiendo de los parámetros de la matriz A , entonces la matriz L se ajusta automáticamente dependiendo de los parámetros del vehículo. Para este comando se usa como parámetros los espacios de estado del sistema, Q es el valor de covarianza del ruido, R el de covarianza del ruido de medición y N el valor de covarianza cruzada de ruido de proceso y medición de valor positivo, que por defecto es 0. Se realizó una serie de pruebas hasta encontrar los valores óptimos los cuales son $Q=10$ y $R=0.1$.

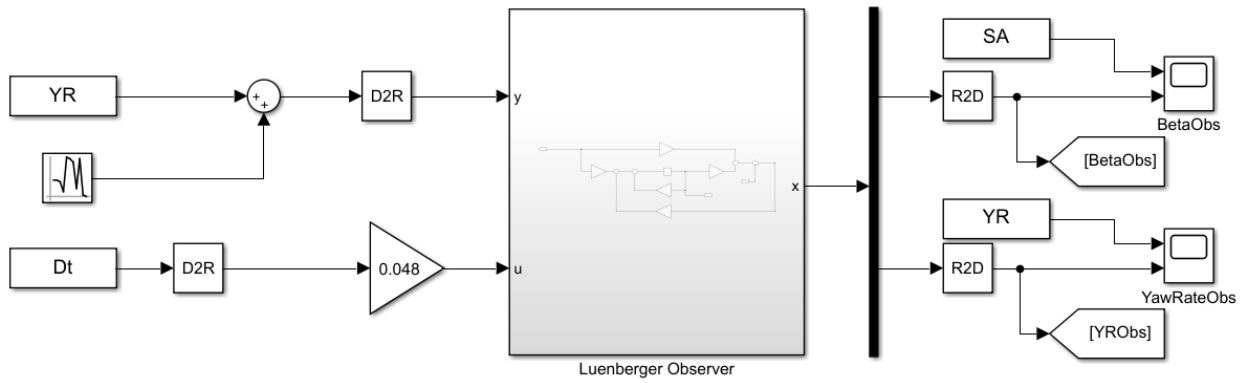


Figura 5-26: Sistema con Observador Luenberger

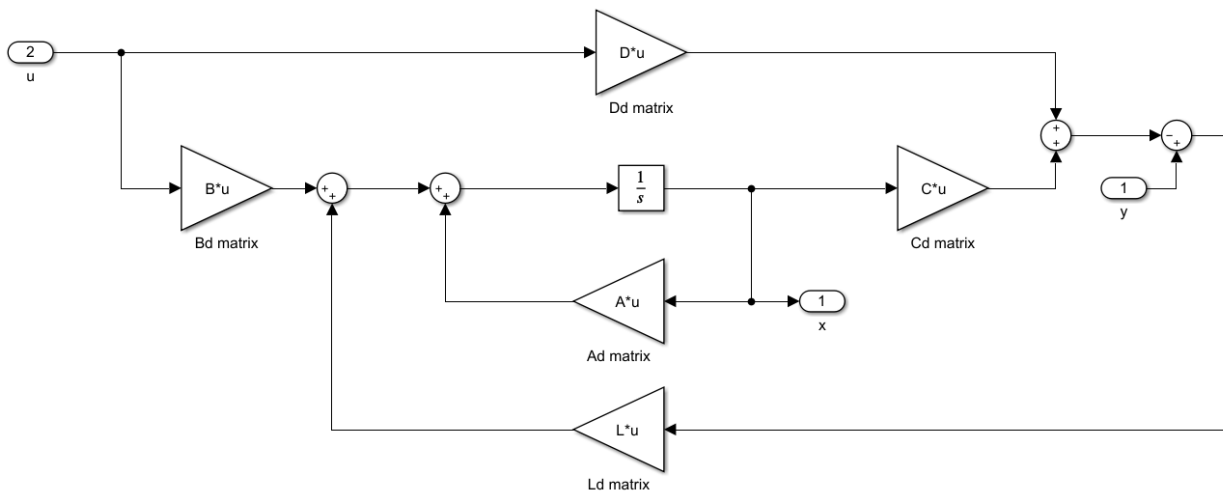


Figura 5-27: Bloque de Observador Luenberger

5.5.1 Simulaciones

5.5.1.1 Giro circular a 40 km/h

Durante esta maniobra el vehículo giro a 15° aproximadamente de manera constante de modo que permanece moviéndose en forma de una circunferencia a velocidad de 40 km/h.

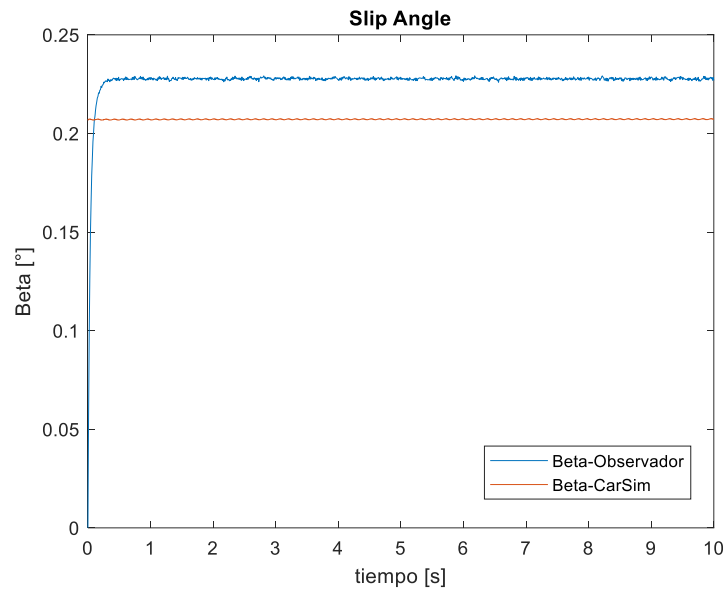


Figura 5-28: Ángulo de deslizamiento lateral, simulación giro circular 40 km/h. Modelo basado en Observador Luenberger.

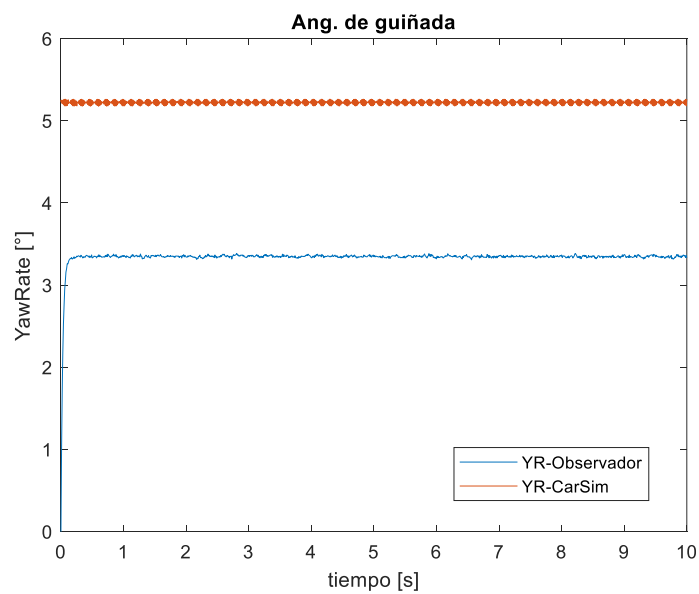


Figura 5-29: Ángulo de guiñada, simulación giro circular 40 km/h. Modelo basado en Observador Luenberger.

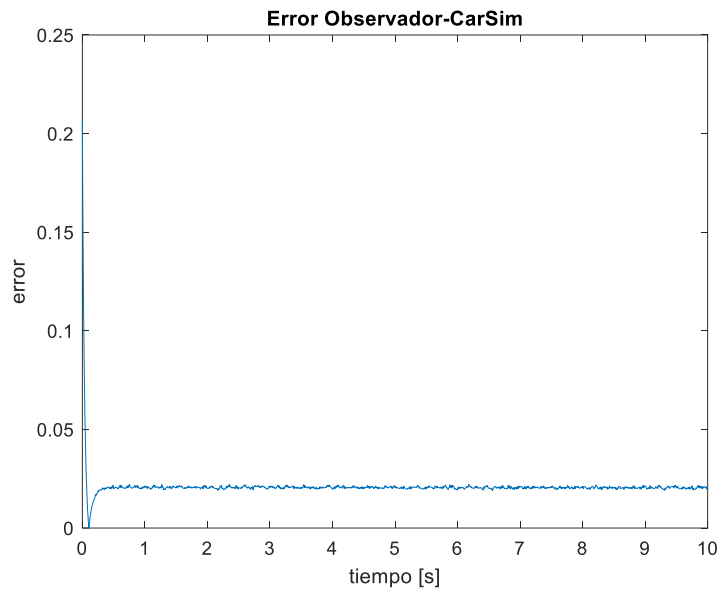


Figura 5-30: Error β Carsim respecto β de la simulación simulación giro circular 40 km/h. Modelo basado en Observador Luenberger.

El error en este caso es un poco mayor al del caso del modelo lineal, pero sigue siendo aceptable.

5.5.1.2 Doble cambio de carril a 120 km/h

Durante esta maniobra el vehículo se encuentra realizando una trayectoria recta a una velocidad de 120 km/h, gira para cambiar de carril, luego gira en sentido contrario para volver al carril inicial.

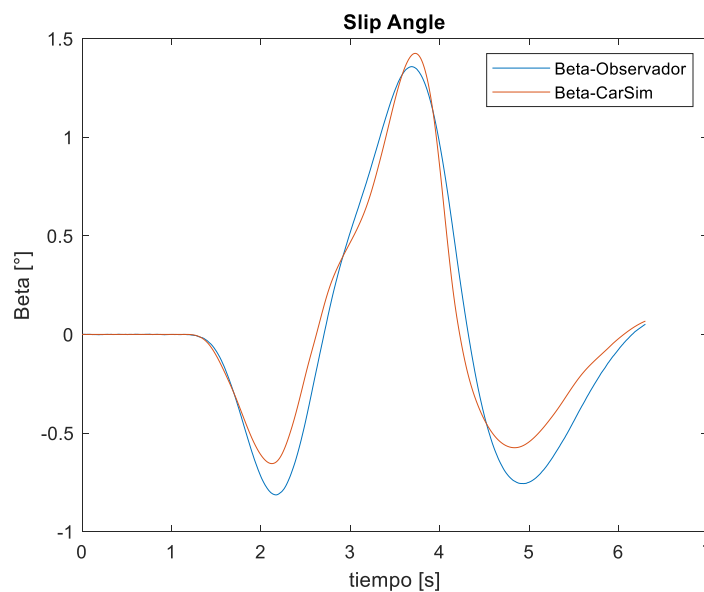


Figura 5-31: Ángulo de desplazamiento lateral, simulación doble cambio de carril a 120 km/h. Modelo basado en Observador Luenberger.

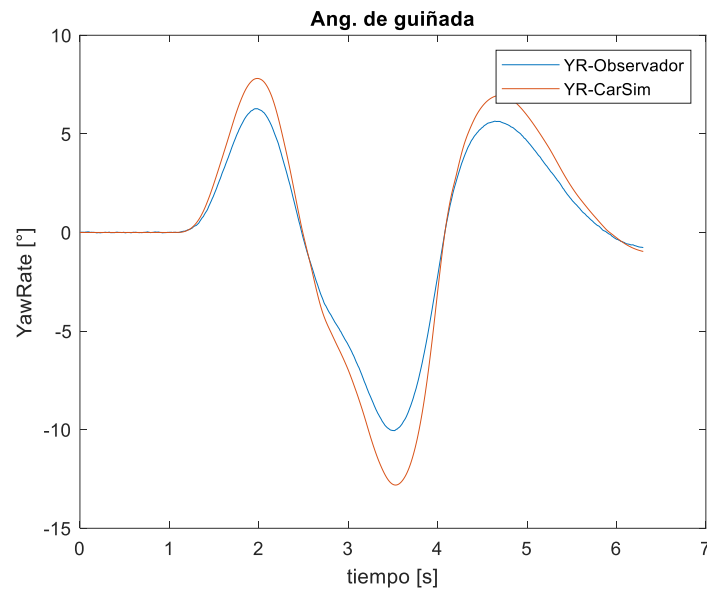


Figura 5-32: Ángulo de guiñada, simulación doble cambio de carril a 120 km/h. Modelo basado en Observador Luenberger.

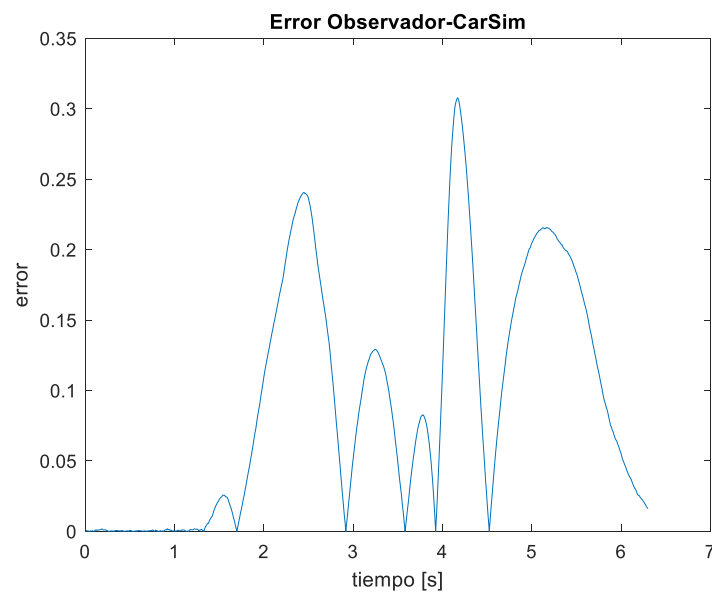


Figura 5-33: Error β Carsim respecto β de la simulación doble cambio de carril a 120 km/h. Modelo basado en Observador Luenberger.

Modelo

El error de la simulación es mínimo, se puede ver en la figura 5-31 el ángulo β estimado es similar al β real de la simulación.

5.5.1.3 Movimiento sinusoidal a 50 km/h

Durante esta maniobra el automóvil se encuentra realizando giros con tal que forma una trayectoria sinusoidal uniforme.

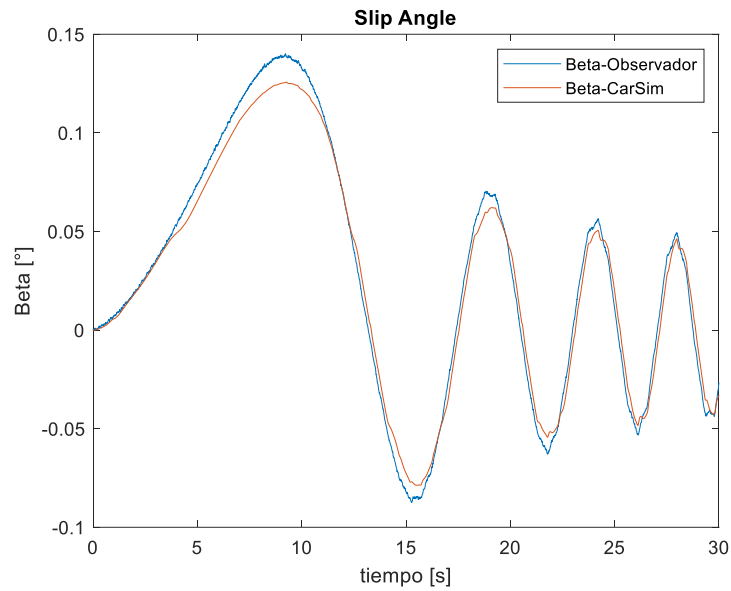


Figura 5-34: Ángulo de desplazamiento lateral, simulación movimiento sinusoidal a 50 km/h. Modelo basado en Observador Luenberger.

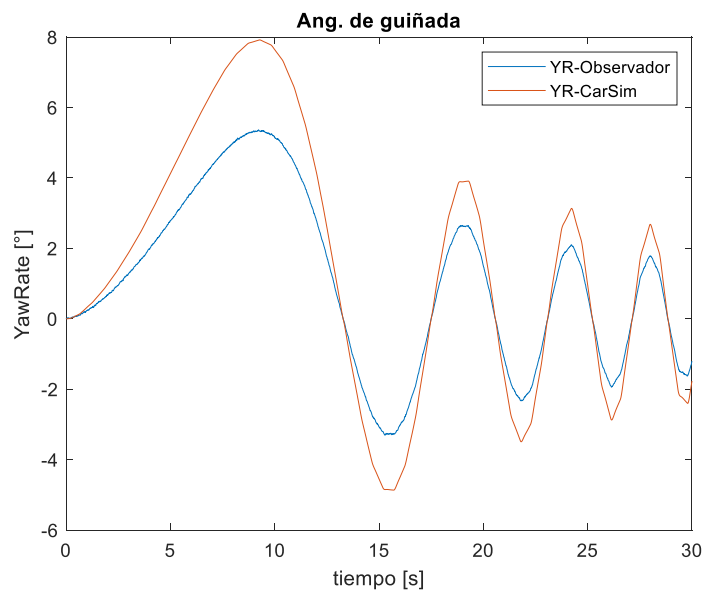


Figura 5-35: Ángulo de guiñada, simulación movimiento sinusoidal a 50 km/h. Modelo basado en Observador Luenberger.

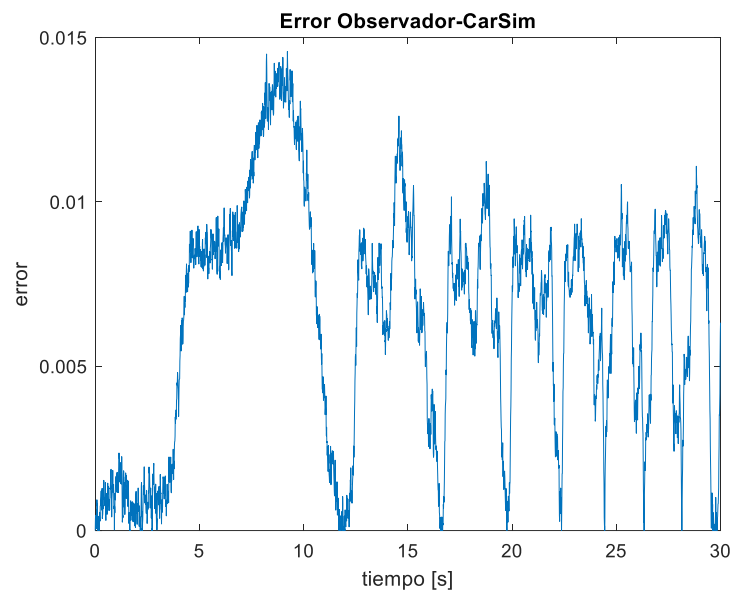


Figura 5-36: Error β Carsim respecto β de la simulación movimiento sinusoidal a 50 km/h. Modelo basado en Observador Luenberger.

La respuesta del modelo de estimación basado en el observador Luenberger es mucho mejor que el modelo lineal ya que estima dependiendo de señales de ángulo de guiñada y de ángulo de giro del volante.

5.6 Modelo basado en Filtro de Kalman

Para modelar el sistema con el Filtro de Kalman se utilizan las matrices discretas de la fórmula 5-3 del modelo lineal del automóvil, en el que las entradas son el ángulo de giro del volante (δ) y el ángulo de guiñada (Υ), que representan las señales que van desde los sensores del vehículo. Y como salidas el ángulo de deslizamiento lateral (β) y el ángulo de guiñada (Υ) de seguimiento.

Las matrices de configuración son G que es una matriz real con tantas filas como A , H es una matriz con tantas filas como C , la matriz Q es para la matriz de covarianza del ruido del proceso de tamaño igual al número de entradas del proceso, R es la matriz de covarianza del ruido de medición, el tamaño de esta matriz es igual el número de salidas y N es la matriz de covarianza cruzada de ruido de proceso y medición de valor positivo. Las matrices de configuración del filtro son: $G = [0.01; 0.3]$, $H=0.5$; $Q=10$; $R=0.1$ y $N=0$. Estos valores se obtuvieron con una serie de pruebas para obtener los valores mas óptimos.

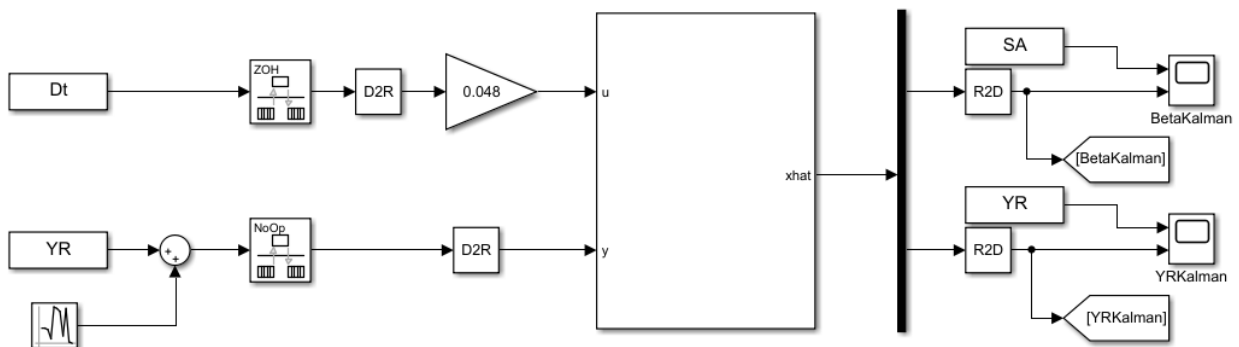


Figura 5-37: Sistema lineal Simulink

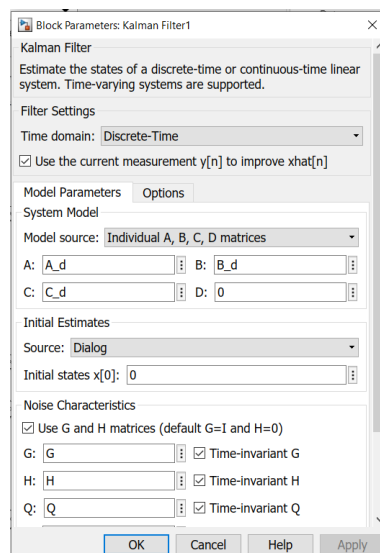


Figura 5-38: Configuración filtro de Kalman

5.6.1 Simulaciones

5.6.1.1 Giro circular a 40 km/h

Durante esta maniobra el vehículo gira a 15° aproximadamente de manera constante de modo que el vehículo permanece moviéndose en forma de una circunferencia a velocidad de 40 km/h.

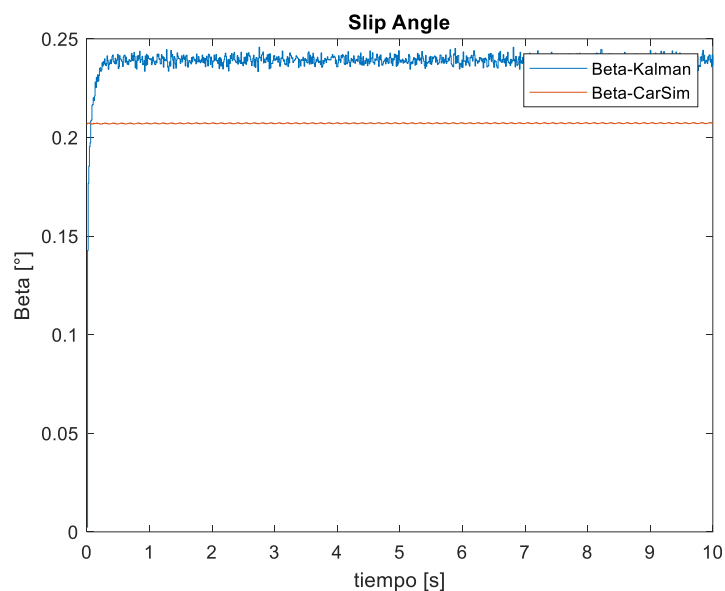


Figura 5-39: Ángulo de desplazamiento lateral, simulación giro circular 40 km/h. Modelo Filtro de Kalman.

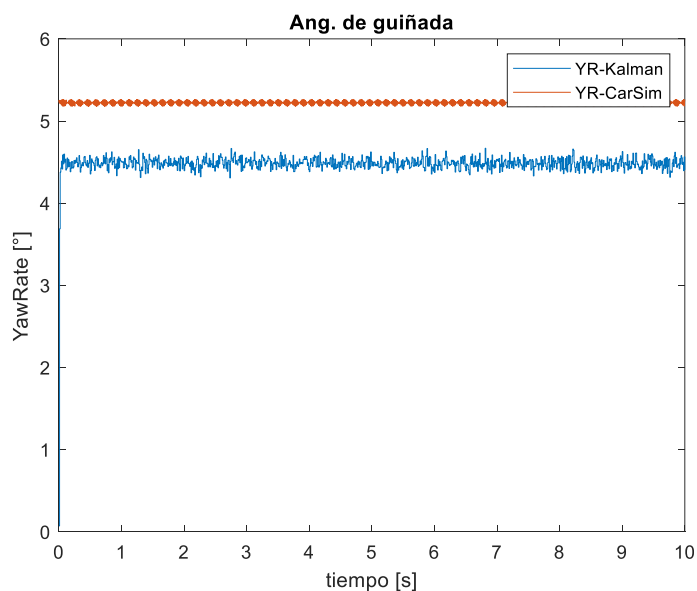


Figura 5-40: Ángulo de guiñada, simulación giro circular 40 km/h. Modelo Filtro de Kalman.

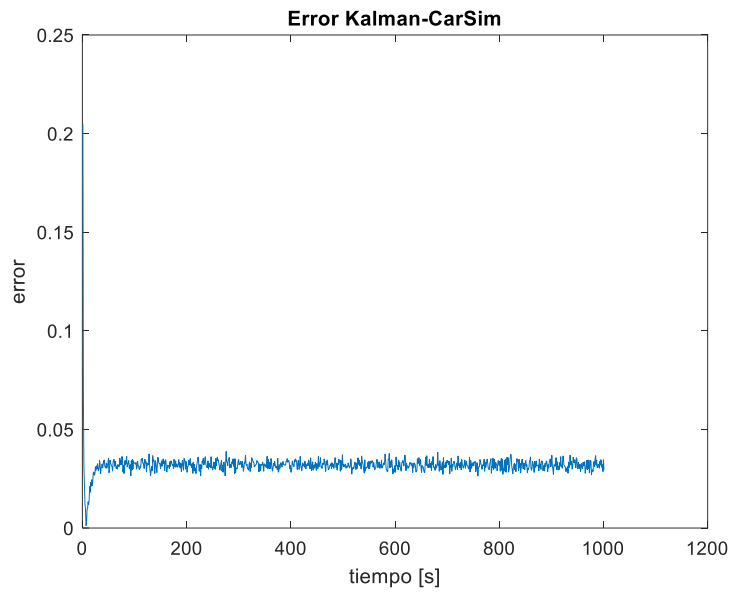


Figura 5-41: Error β Carsim respecto β de la simulación giro circular 40 km/h. Modelo Filtro de Kalman.

El error del ángulo β estimado respecto al ángulo β de CarSim es mínimo, esta en el rango de lo aceptable.

5.6.1.2 Doble cambio de carril a 120 km/h

Durante esta maniobra el vehículo se encuentra realizando una trayectoria recta a una velocidad de 120km/h, gira para cambiar de carril, luego gira en sentido contrario para volver al carril inicial.

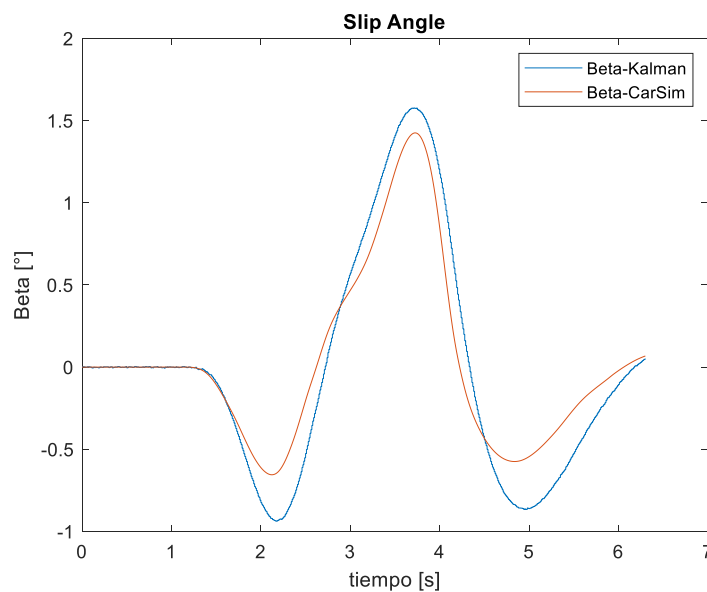


Figura 5-42: Ángulo de desplazamiento lateral, simulación doble cambio de carril a 120 km/h. Modelo Filtro de Kalman.

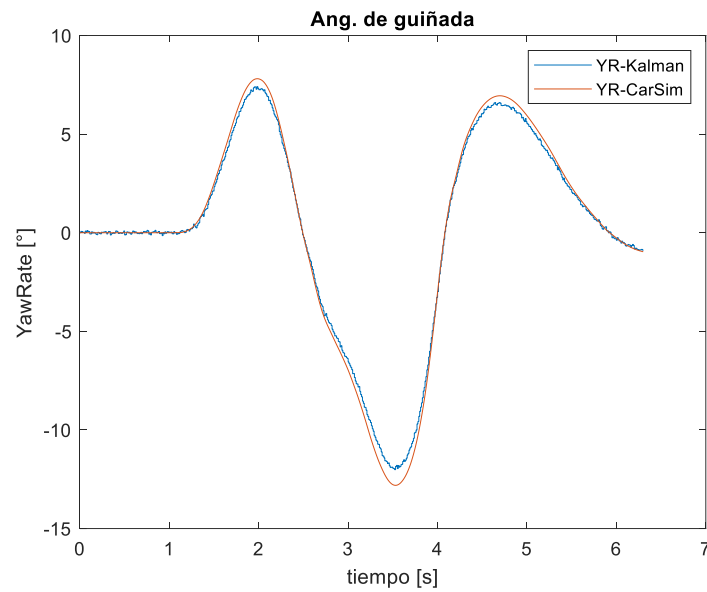


Figura 5-43: Ángulo de guiñada, simulación doble cambio de carril a 120 km/h. Modelo Filtro de Kalman.

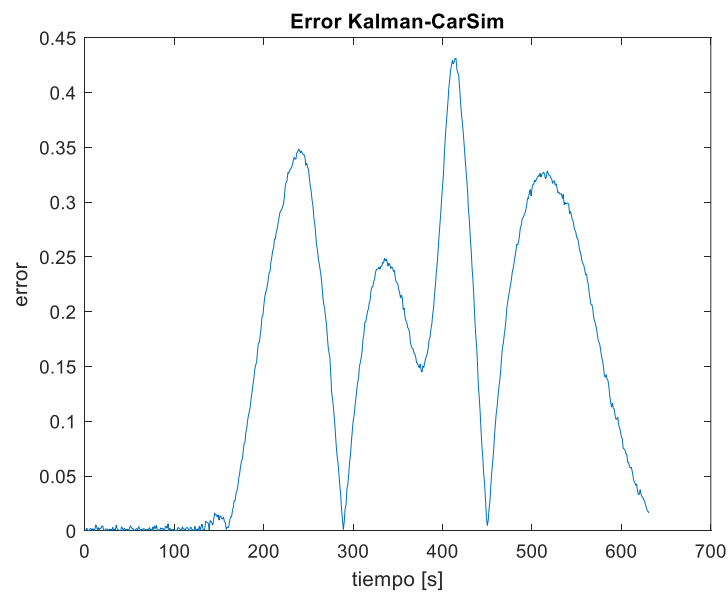


Figura 5-44: Error β Carsim respecto β de la simulación doble cambio de carril a 120 km/h. Modelo Filtro de Kalman.

Como se puede observar en la figura 5-42 el ángulo β estimado es bastante similar al β real, el error es mínimo.

5.6.1.3 Movimiento sinusoidal a 50 km/h

Durante esta maniobra el vehículo se encuentra realizando giros con tal que forma una trayectoria sinusoidal uniforme.

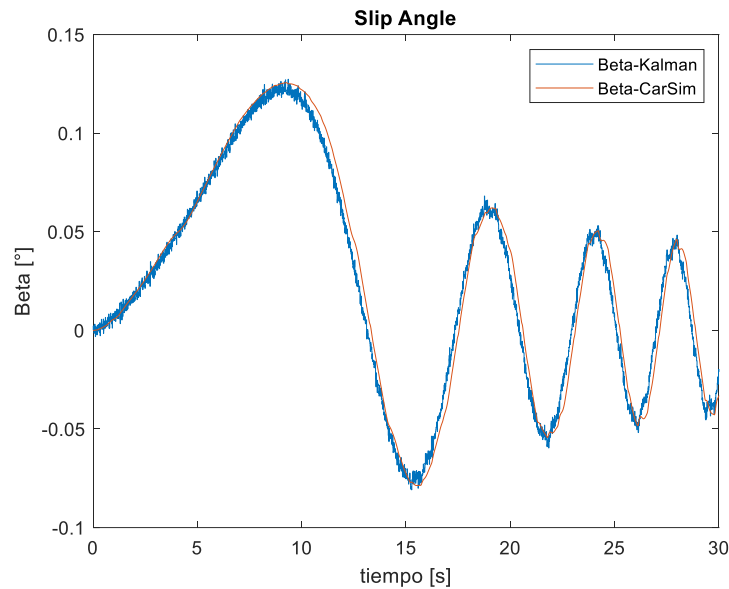


Figura 5-45: Ángulo de desplazamiento lateral, movimiento sinusoidal. a 50km/h. Filtro de Kalman.

El ángulo β estimado es muy parecido al de la del simulador CarSim, tiene un error muy pequeño y el seguimiento es similar.

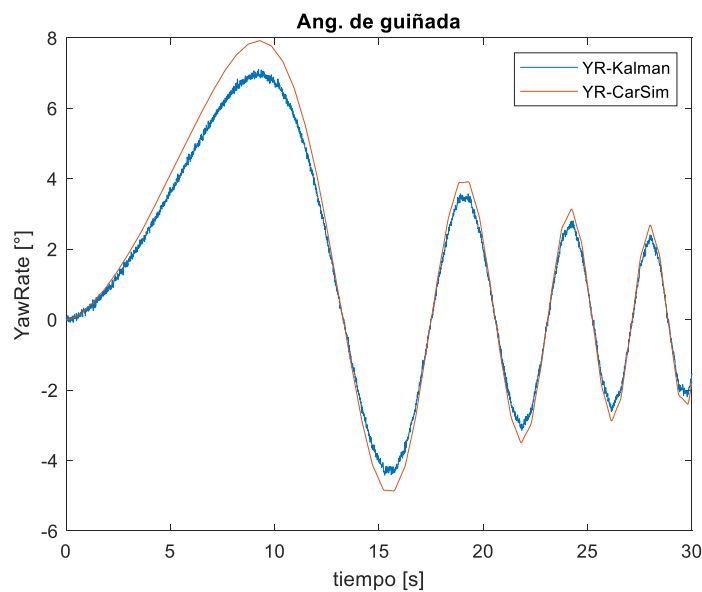


Figura 5-46: Ángulo de guiñada, movimiento sinusoidal. a 50km/h. Filtro de Kalman.

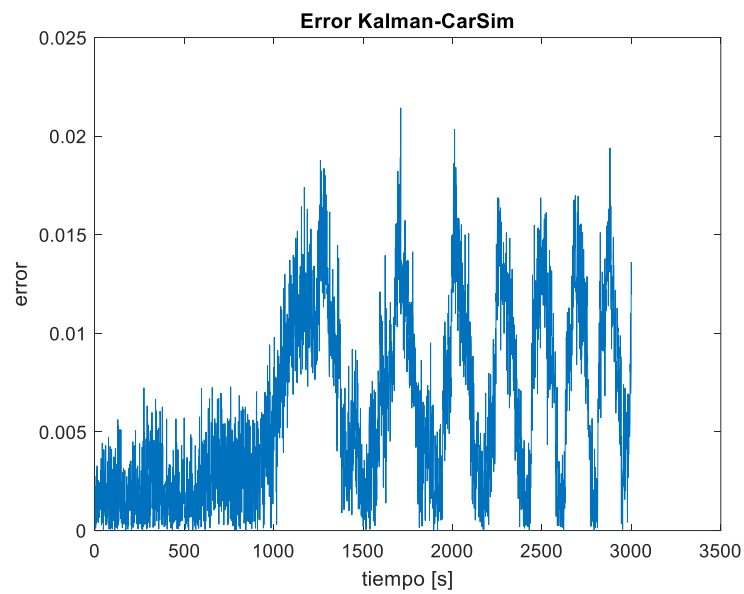


Figura 5-47: Error β Carsim respecto β de la simulación movimiento sinusoidal 50 km/h. Modelo Filtro de Kalman.

El error en este caso es bajo y se puede apreciar en la figura 5-45 que el ángulo β estimado es próximo al de la simulación de CarSim.

5.7 Modelo basado en Redes Neuronales

Para efectuar la red neuronal empleamos el lenguaje Python. Utilizamos: la librería torch que es un software gratuito y de código abierto, que proporciona funciones de alto nivel para redes neuronales profundas construidas en un sistema de diferenciación automática en donde el módulo facilita la definición de gráficos y la toma de gradientes. NumPy, que es la librería que da soporte para crear vectores y matrices grandes multidimensionales, junto con una gran colección de funciones matemáticas de alto nivel para operar con ellas. Pandas es la librería que define nuevas estructuras de datos basadas en los arrays de la librería NumPy, pero con nuevas funcionalidades, permite leer los archivos de datos .csv y reordena estos datos. Matplotlib es la librería para la generación de gráficos a partir de datos contenidos en listas o arrays.

```
import torch.nn as nn
import torch.nn.functional as F
import torch
import torch.optim as optim
import numpy as np
import pandas
import matplotlib.pyplot as plt
```

Figura 5-48: Librerías de la red neuronal.

Se creó una red neuronal como se indicó en la figura 4-7, con 4 neuronas en la capa de entrada, para la velocidad del vehículo (V_x), ángulo de dirección (δ), aceleración lateral (A_y), y ángulo de guiñada (γ) y una neurona en la capa de salida para el ángulo de deslizamiento lateral del vehículo (β).

En las capas ocultas se probó con una capa oculta de 10 neuronas, 2 capas ocultas de 10 neuronas cada una y otra de 3 capas ocultas con 10 neuronas cada una. La que dio mejores resultados fue la de tres capas ocultas.

El aprendizaje de la red se hizo con las pruebas mencionadas en los puntos anteriores, para tener unos mejores resultados y ser mucho más precisos se debería entrenar a la red neuronal con una gran cantidad de experimentos, ya que la red neuronal entre mas sea entrenada los resultados de la estimación serán mucho más exactos.

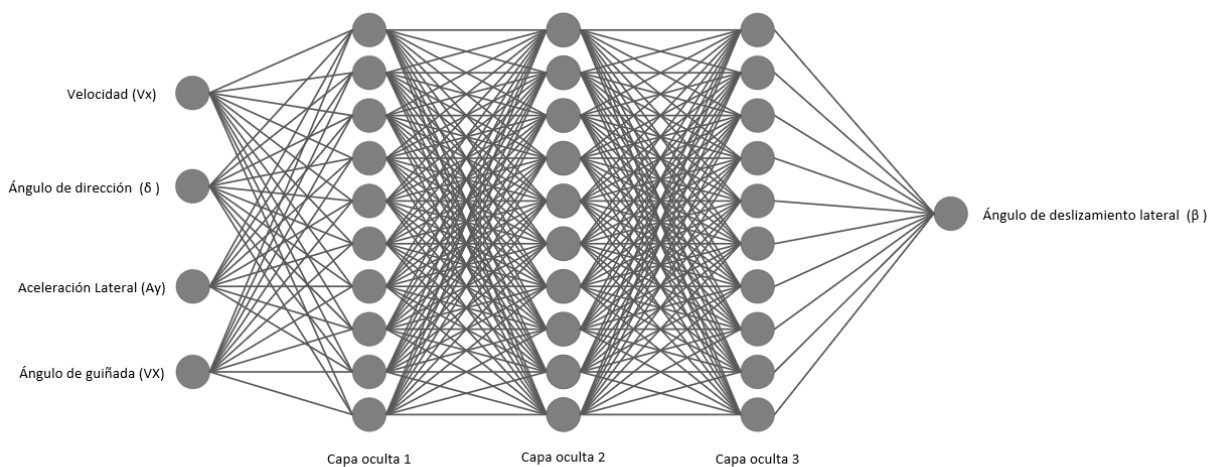


Figura 5-49: Red neuronal implementada para la estimación del ángulo de deslizamiento lateral.

5.7.1 Experimento

Se utilizaron los datos de los tres experimentos para entrenar a la red neuronal y se probó con la de cambio de carril a 50 km/h. En la teoría de algunos artículos de investigación recomiendan que el 90% de los datos sea para entrenar a la red y el 10 % para presentar resultados.

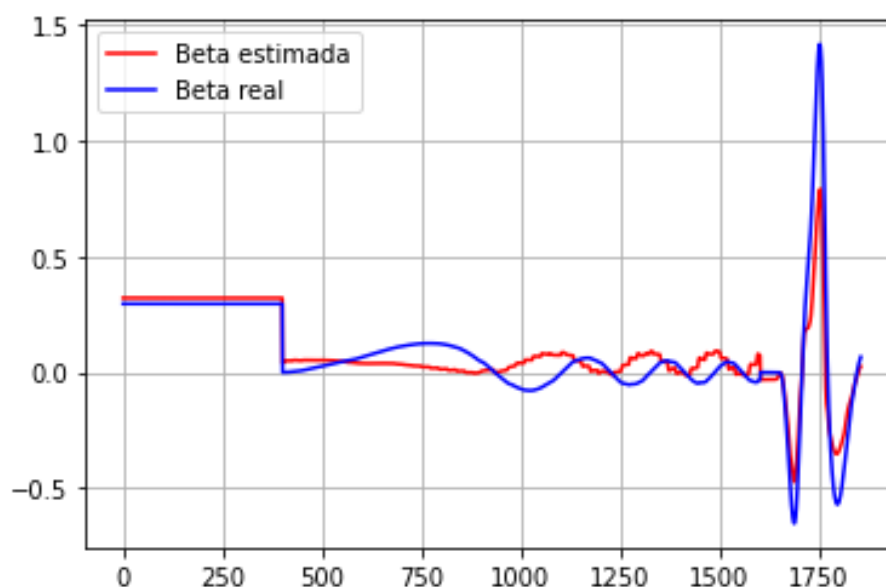


Figura 5-50: Entrenamiento de la red neuronal.

Respuesta de la red neuronal a la prueba de doble cambio de carril a 120km/h.

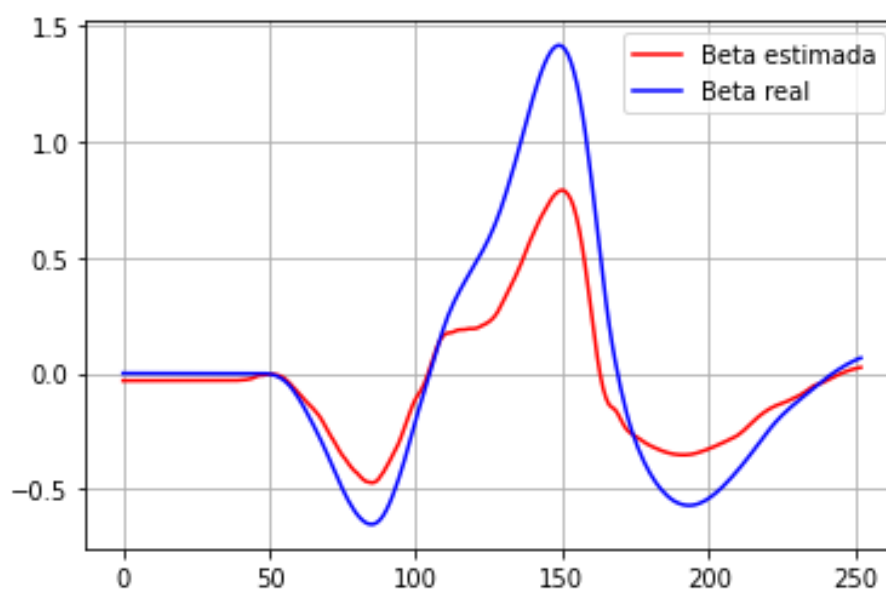


Figura 5-51: Estimación de β de la red neuronal a la prueba de doble cambio de carril a 120km/h.

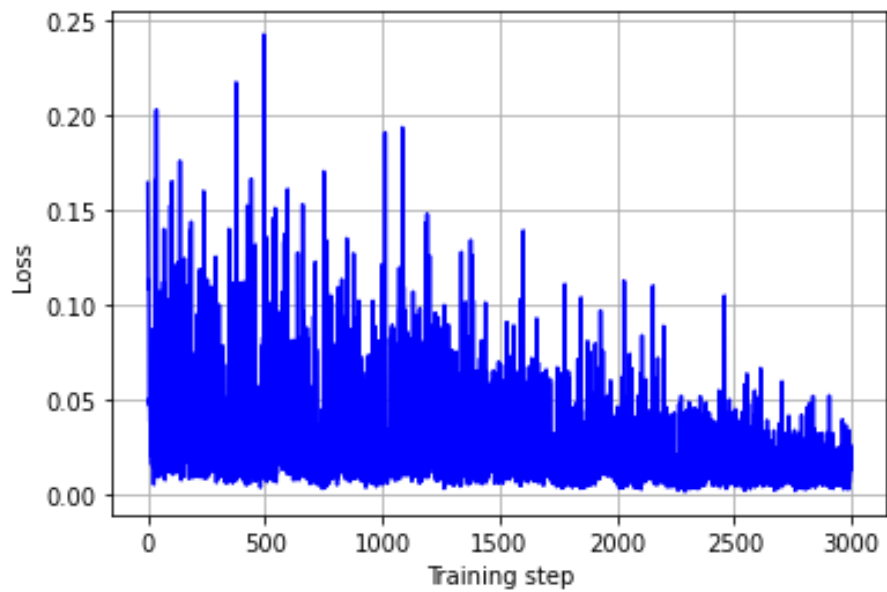


Figura 5-52: Error entre el β de CarSim y el β a la salida de la red neuronal.

El error en este primer experimento es bajo, pero el ángulo β estimado no es similar al β de la simulación de CarSim. Para mejorar este resultado se necesita mas entrenamiento, con muchos experimentos mas.

Se realiza otro experimento con los mismos datos.

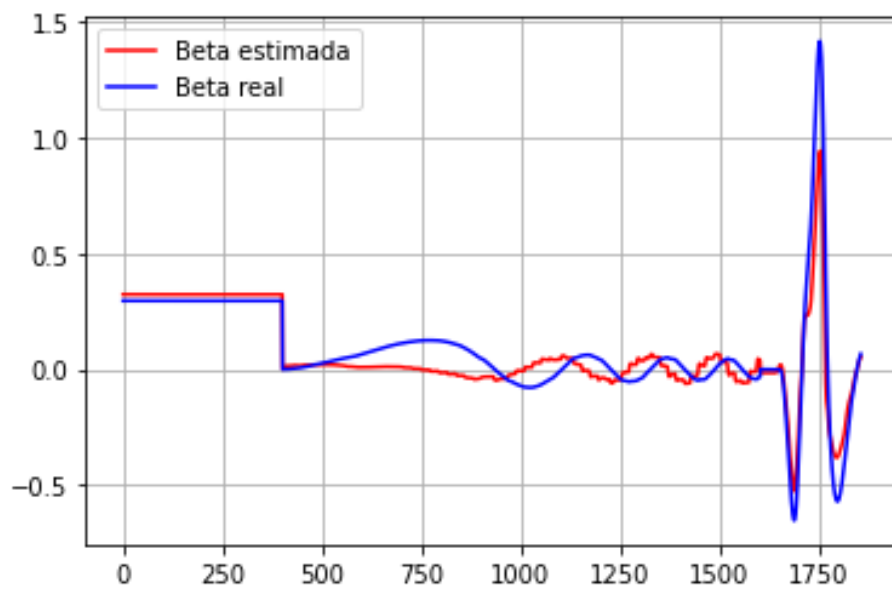


Figura 5-53: Entrenamiento de la red neuronal.

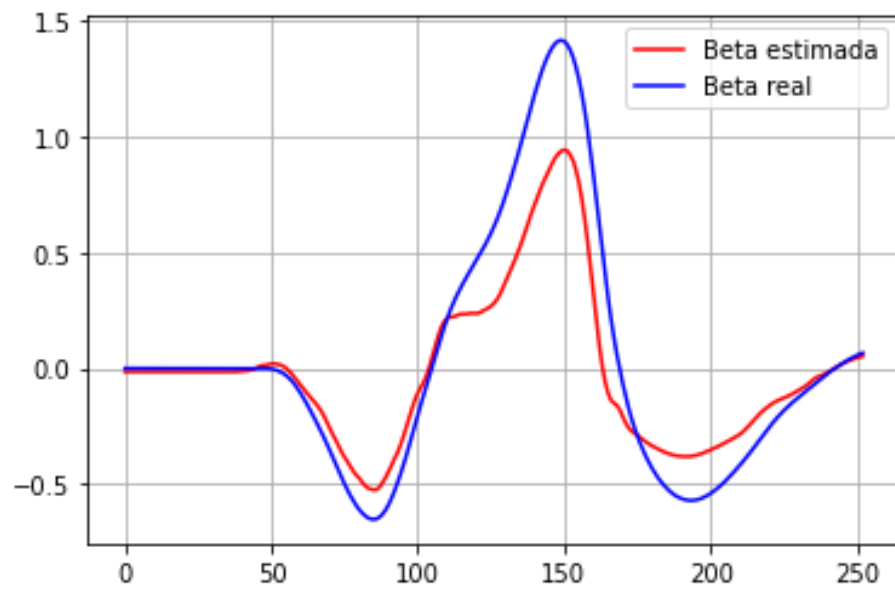


Figura 5-54: Estimación de β de la red neuronal a la prueba de doble cambio de carril a 120 km/h.

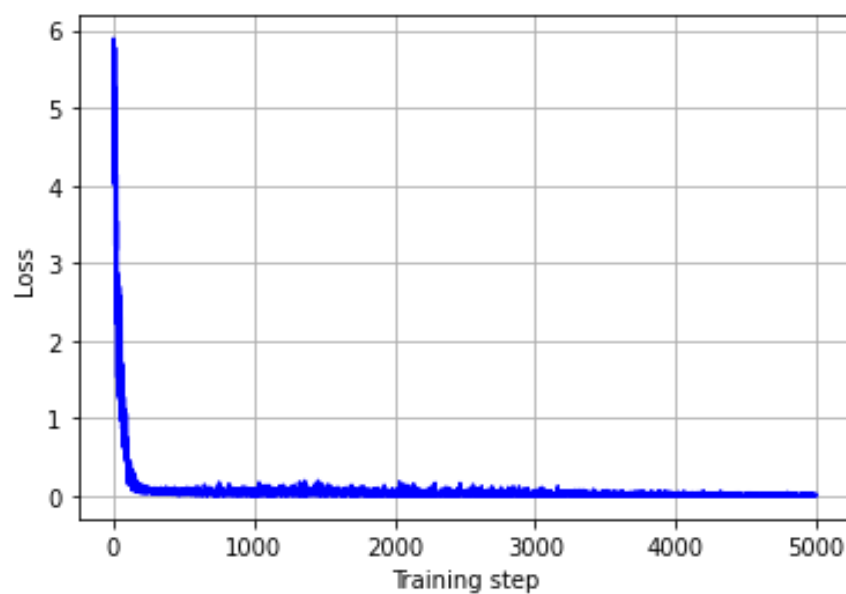


Figura 5-55: Error entre el β de CarSim y el β a la salida de la red neuronal.

Como se puede observar el experimento de la red neuronal utiliza datos de los tres experimentos para poder estimar el ángulo de otro experimento. El rendimiento de la red neuronal se puede mejorar con mucho mas entrenamiento.

Resumen

Los métodos en general tienen un rendimiento similar, ya que dan un error bastante aceptable. El método con menor error es la red neuronal ya que toma en cuenta un modelo no lineal, con velocidad longitudinal variable, del vehículo. Los métodos con Observador Luenberger y con filtro de Kalman están basados en el modelo lineal por lo que no pueden ser puestos a prueba con velocidad longitudinal variable.

La desventaja de la red neuronal es que necesita datos de muchos experimentos para el entrenamiento y optimizar la estimación del ángulo de deslizamiento lateral además de que el tiempo de procesamiento es alto y no serviría para aplicaciones en las que se estime en tiempo real.

Tabla 5–2 Error medio en la estimación del ángulo de deslizamiento lateral

	Prueba de manejo circular 40km/h	Prueba de manejo de doble cambio de carril a 120km/h	Prueba de manejo de sinusoidal 50km/h
Modelo lineal	0.004945	0.04524	0.004021
Observador Luenberger	0.02183	0.06351	0.006034
Filtro de Kalman	0.0319	0.1529	0.006118
Red Neuronal	-	0.032	-

Tabla 5–3 Error máximo en la estimación del ángulo de deslizamiento lateral

	Prueba de manejo circular 40km/h	Prueba de manejo de doble cambio de carril a 120km/h	Prueba de manejo de sinusoidal 50km/h
Modelo lineal	0.2071	0.3105	0.008826
Observador Luenberger	0.2071	0.3078	0.01458
Filtro de Kalman	0.2048	0.4313	0.02143
Red Neuronal	-	0.2	-

6 EXPERIMENTO CON DATOS DEL VEHÍCULO FOX

6.1 Introducción

En esta parte se van a realizar las pruebas con los datos de los experimentos con el vehículo FOX, del TFG “Control de la dinámica de vehículos eléctricos con motores en las ruedas” de Juan Carlos Fraile de la Universidad de Sevilla. [24]

El FOX es un vehículo eléctrico desarrollado en la Universidad de Sevilla que presenta dos principales novedades frente a los modelos convencionales:

- Sistema de alimentación eléctrica que permite combinar el uso de baterías y una pila de hidrógeno.
- Tracción integral a las cuatro ruedas mediante la inclusión de 4 motores intrarueda, ofreciendo mayor aptitud que los vehículos que presentan como máximo un motor por eje, para el desarrollo de los sistemas de tracción y estabilidad.



Figura 6-1: Vehículo FOX.

El vehículo FOX dispone de una unidad de medición inercial (IMU) que proporciona las medidas del vehículo. Esto lo realiza a través de los siguientes sensores:

- Acelerómetro: Proporciona la aceleración lineal y angular del FOX
- Magnetómetro: Proporciona los ángulos de Euler referenciados a un sistema de ejes local.
- Giróscopo: Proporciona Velocidad angular del vehículo.
- GPS: Posición y velocidad referenciados a un sistema de coordenadas no locales (Ejes NED).

Además de la IMU el FOX cuenta con otros sensores:

- Potenciómetros lineales en la amortiguación utilizados para conocer las fuerzas verticales que actúan sobre cada rueda.

- Potenciómetros lineales en los pedales utilizados para conocer la posición o recorrido de cada pedal.
- Potenciómetro rotatorio en el volante empleado para conocer el ángulo de giro del volante.

Los ángulos de Euler proporcionados por el magnetómetro son utilizados para convertir la velocidad lineal medida por el GPS a velocidad longitudinal y lateral, y con esto calcular el ángulo de deslizamiento lateral, con la fórmula 2-1.

6.2 Parámetros del FOX

Los parámetros del FOX para las simulaciones se resumen en la tabla 6-1.

Tabla 6-1 Parámetros del vehículo FOX

Parámetro	Nombre del parámetro	Valor [unidad]
m	Masa del vehículo	400.24 [kg]
I_{zz}	Momento de inercia del vehículo alrededor del eje z	1047.5 [kg m ²]
C_{af}	Rigidez a deslizamiento lateral en la rueda delantera	47000 [N/rad]
C_{ar}	Rigidez a deslizamiento lateral en la rueda trasera	53000 [N/rad]
l_f	Distancia entre el centro de gravedad y el eje delantero	1.3 [m]
l_r	Distancia entre el centro de gravedad y el eje trasero	1.22 [m]

La prueba realizada con el vehículo FOX sigue la trayectoria de la figura 6-2, en la que el ángulo de giro y la velocidad está alrededor de las figuras 6-3 y 6-4 respectivamente.

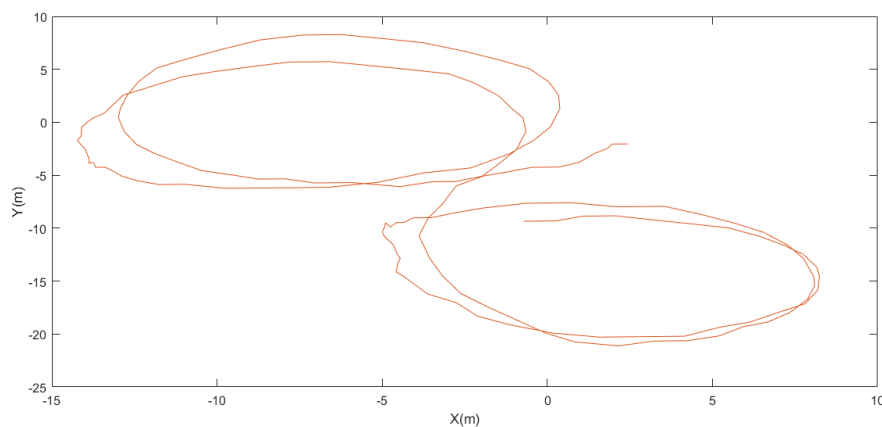


Figura 6-2: Trayectoria, prueba con FOX.

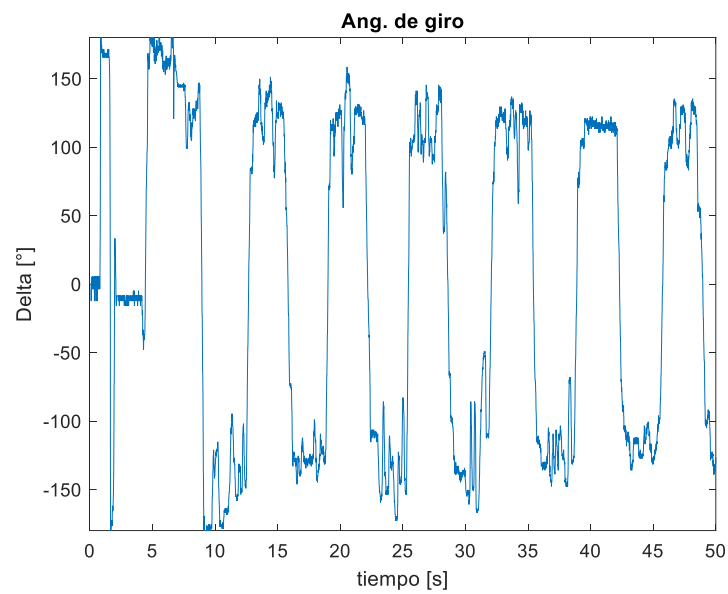


Figura 6-3: Ángulo de giro, prueba con FOX.

La velocidad longitudinal es aproximadamente 5 m/s que será utilizado en como parámetro para las simulaciones del modelo lineal, modelo del observador Luenberger y con el modelo utilizando filtro de Kalman.

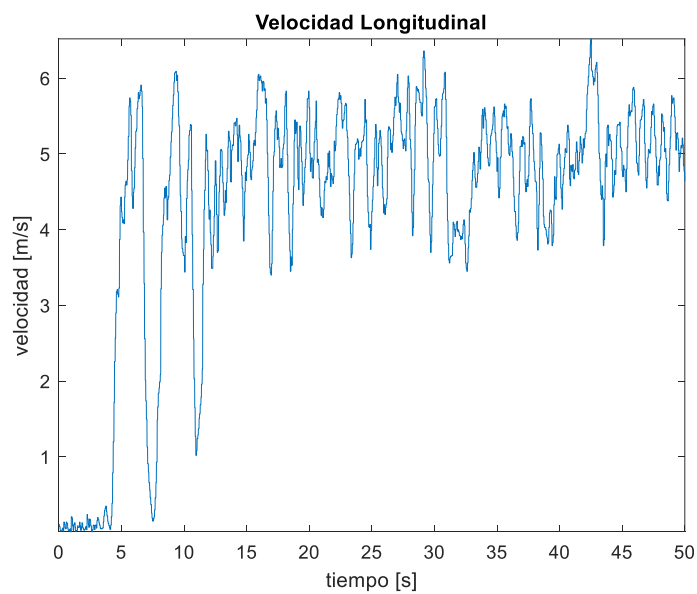


Figura 6-4: Velocidad longitudinal del vehículo FOX.

6.3 Prueba con el modelo Lineal

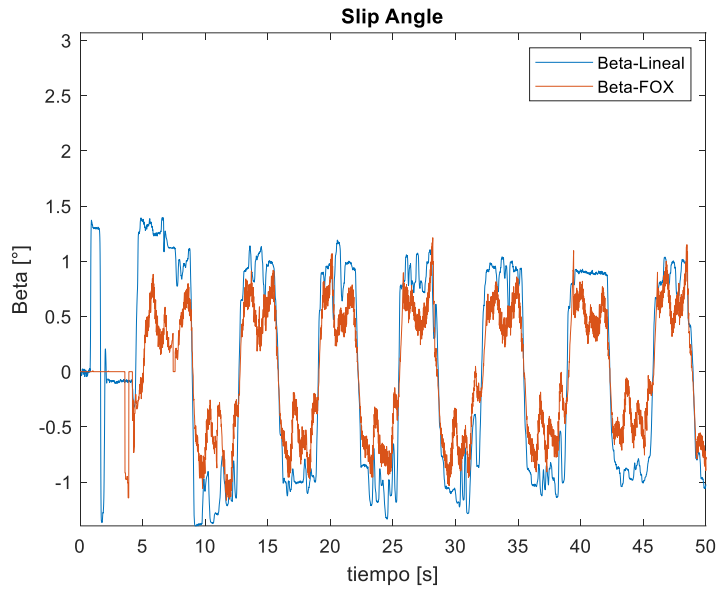


Figura 6-5: Ángulo de deslizamiento lateral, con modelo lineal.

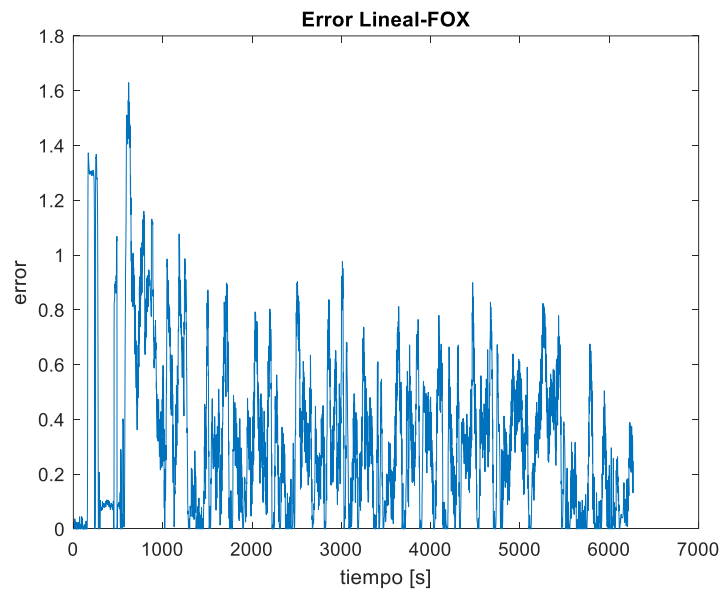


Figura 6-6: Error entre β real y β estimado con modeo lineal.

Como se puede observar en la figura 6-5 el β estimado con el modelo lineal es bastante aproximado al β calculado en el experimento real. El error es producido porque la velocidad real no es constante, sino que tiene un rango de variación y en la matriz de parámetros es una constante.

6.4 Prueba con el modelo basado en Observador Luenberger

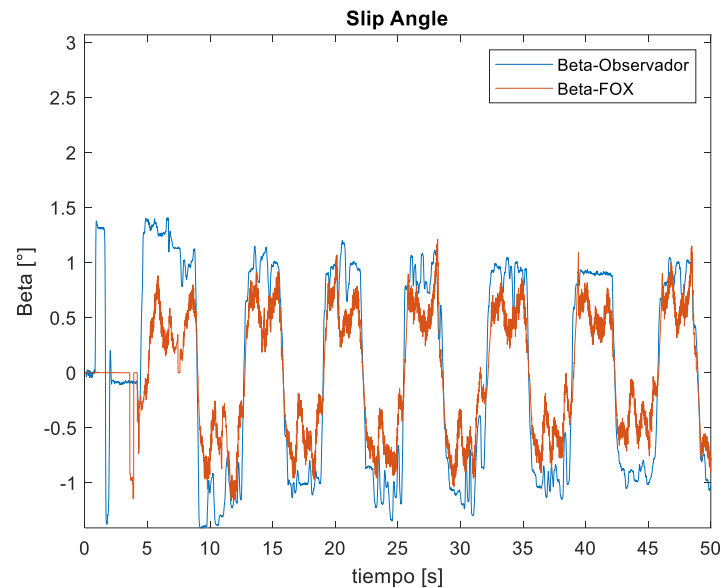


Figura 6-7: Ángulo de deslizamiento lateral, con modelo basado en observador Luenberger.

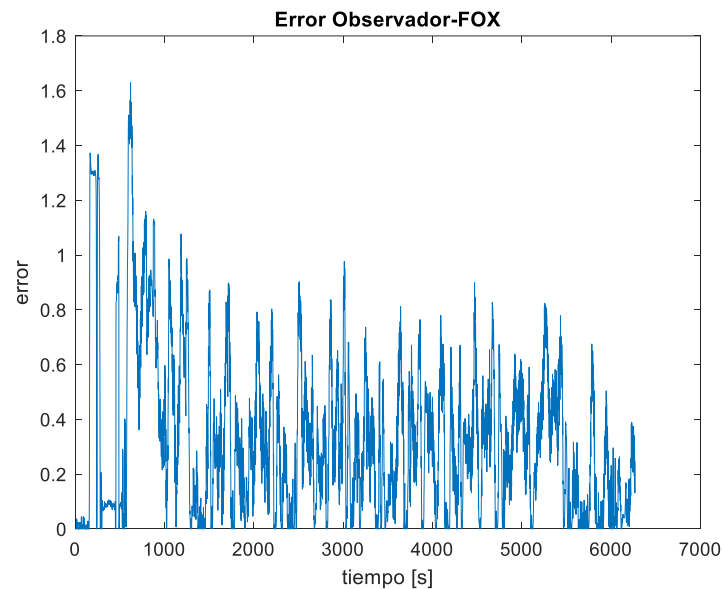


Figura 6-8: Error entre β real y β estimado con modeo basado en observador Luenberger.

El β estimado con el modelo con observador Luenberger es muy próximo al β calculado en el experimento real. El error es producido porque la velocidad real no es constante, sino que tiene un rango de variación y en la matriz de parámetros es una constante.

6.5 Prueba con el modelo basado en Filtro de Kalman

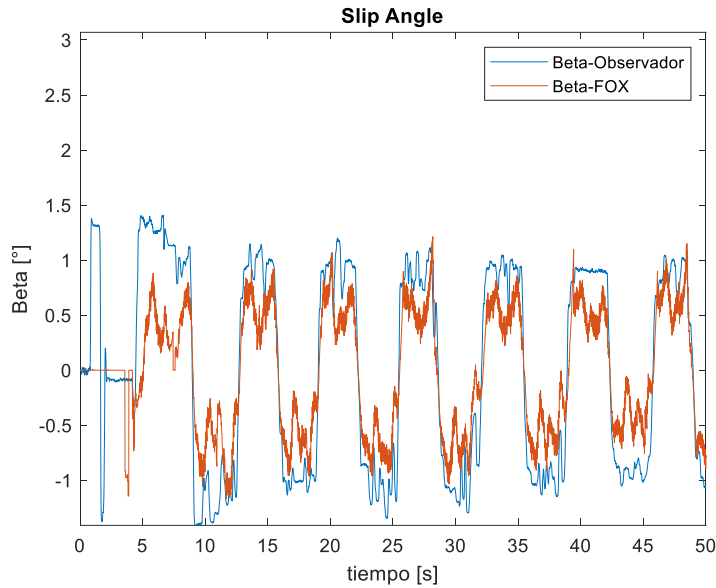


Figura 6-9: Ángulo de deslizamiento lateral, con modelo basado en filtro de Kalman.

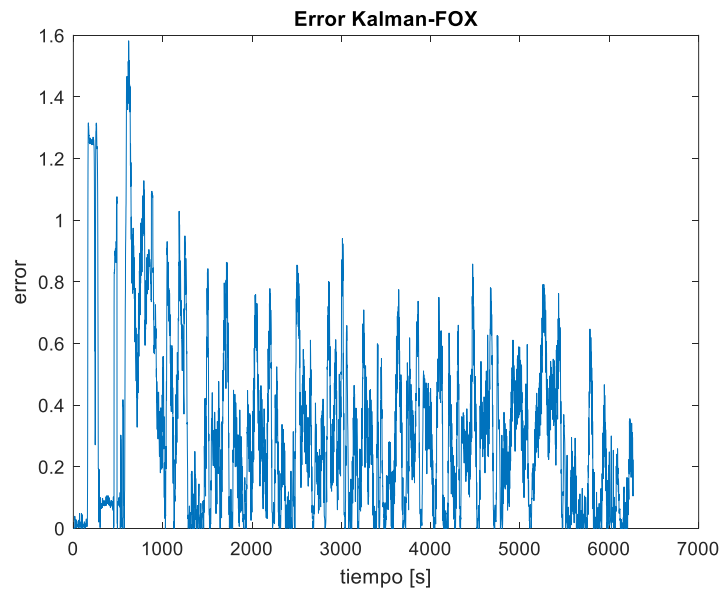


Figura 6-10: Error entre β real y β estimado con modeo basado en filtro de Kalman

El β estimado con el modelo con flitro de Kalman es similar al β calculado en el experimento real. El error es producido porque la velocidad real no es constante, sino que tiene un rango de variación y en la matriz de parámetros es una constante. Este es el que menor error tiene entre los modelos que utilizan el modelo de la bicicleta como matriz de parámetros.

6.6 Prueba con la red neuronal

Para el entrenamiento de la red neuronal se utilizan todos los datos de los experimentos anteriormente mencionados y se hace la prueba con un tramo del experimento con el vehículo FOX.

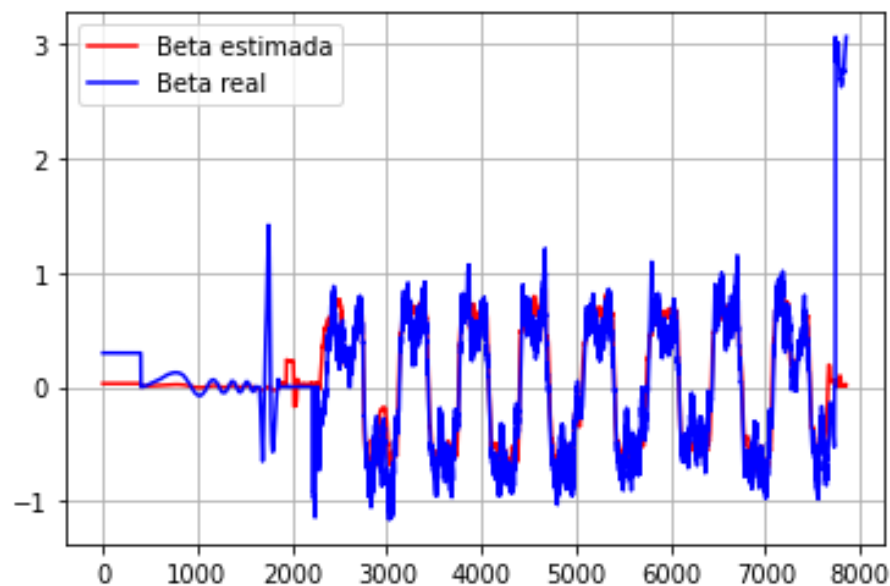


Figura 6-11: Entrenamiento de la red neuronal.

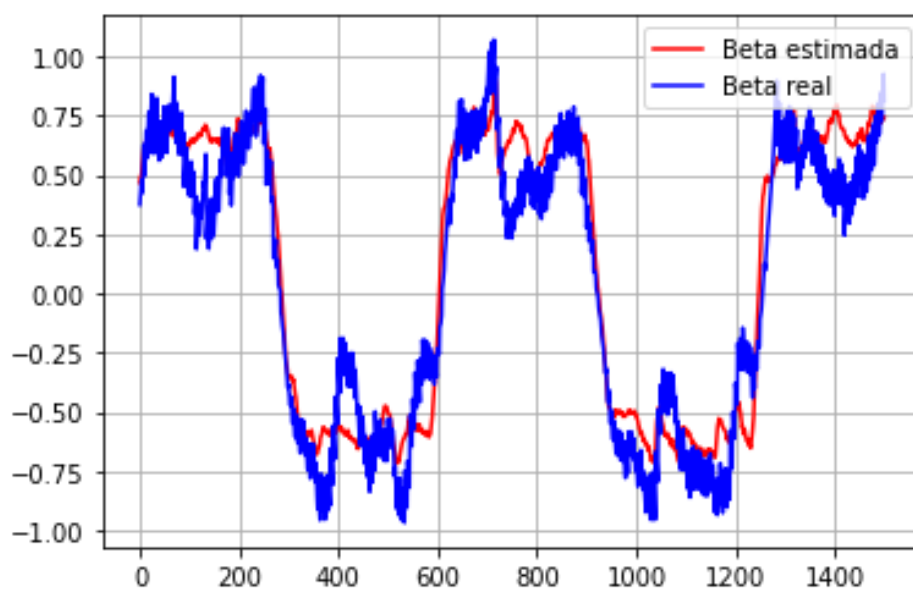


Figura 6-12: Estimación de β de la red neuronal.

El β estimado con la red neuronal es bastante próxima al β real del experimento, ya que la red neuronal toma como entrada la velocidad longitudinal, y en este caso al ser una velocidad que no es constante la red neuronal realiza una buena estimación del β .

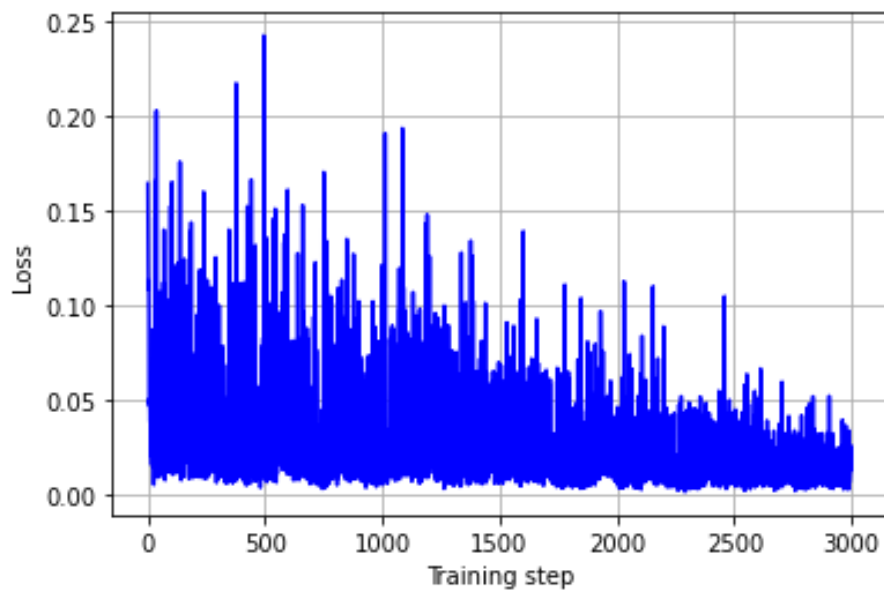


Figura 6-13: Error entre la β estimada y la β de la prueba del vehículo FOX.

En estas pruebas con datos de experimentos reales se puede apreciar que el de mejor rendimiento es la red neuronal, ya que en su diseño toma en cuenta 4 entradas y además está mas entrenada.

7 CONCLUSIONES

7.1 Conclusiones

En el presente Trabajo de Fin de Máster se han realizado 4 modelos con la finalidad de encontrar un modelo para estimar el ángulo de deslizamiento lateral de un vehículo en función de la velocidad longitudinal, el ángulo de guiñada y el ángulo de dirección. También se ha realizado un acercamiento al ángulo de deriva de un vehículo tratando de entender qué es, cómo se forma, de qué variables depende y que estrategias son seguidas para su medición y estimación.

Los modelos para la estimación vistos se basan en el modelo de la bicicleta, tanto para sistema continuo como sistema discreto. El que logra una estimación mas cercana es el filtro de Kalman ya que toma en cuenta la señal del sensor y la del modelo. En estos la velocidad es una entrada de valor constante ya que por este motivo la estimación tiene un error mayor a la de la red neuronal que la considera como entrada variable.

La red neuronal es la que mejor estima, cuando tiene un buen entrenamiento, la única desventaja es que se necesitan una gran cantidad de datos de experimentos y simulaciones para poder entrenarla y la estimación sea lo mas cercana a la realidad. La ventaja de la red neuronal es además que al tener como entrada la velocidad, esta puede ser variable y no una velocidad constante como en el modelo de la bicicleta utilizado para los otros modelos de estimación.

Como trabajos futuros, se puede utilizar estos modelos como base para métodos combinados para poder mejorar la estimación del ángulo de deslizamiento lateral.

REFERENCIAS

- [1] J. Colón, «Diseño de un sistema de estimación del ángulo de deriva de un vehículo automóvil,» 2017.
- [2] J. S. Chawla, *Estimation of Side Slip in Vehicles*, Bombay, 2006.
- [3] R. Rajamani, *Vehicles Dynamic and Control*, 2006.
- [4] Y. Fukada, «Slip- angle estimation for vehicle stability control.,» *Vehicle System Dynamics.*, vol. 32, pp. 375-388, 1999.
- [5] B. L. M. G. Daniel Chindamo, *On the Vehicle Sideslip Angle Estimation: A literature review of methods, models, and innovations*, 2018.
- [6] Håvard Fjær, Lars Imsland, T. A. Johansen, J. Kalkkuhl y A. Suissa, «Vehicle sideslip estimation.,» 2009.
- [7] Moustapha Doumiati, Alessandr Correa Victorino y Ali Charara, «Onboard real-time estimation of vehicle lateral tire–road forces and sideslip angle.,» *IEEE*, 2010.
- [8] H. Kim y J. Ryu, «Sideslip angle estimation considering short-duration longitudinal velocity variation.,» 2011.
- [9] Jihan Ryu, Eric J. Rossetter y Christian Gerdes, «Vehicle Sideslip and Roll Parameter Estimation using GPS.,» 2002.
- [10] L. C. X. X. Te Chen, *Sideslip Angle Fusion Estimation Method of an Autonomous Electric Vehicle Based on Robust Cubature Kalman Filter with Redundant Measurement Information*, 2019.
- [11] P. P. ., N. V. Nguyen, *Estimation of Vehicle Dynamics States Using Luenberger Observer*, Danang city, Vietnam, 2019.
- [12] B. Chen y F. Hsieh, «h. Sideslip angle estimation using extended Kalman filter. Vehicle System Dynamics.,» *Supplement*, n° 353-364, 2008.
- [13] Y. Huang, *Estimation of Sideslip Angle Based on Extended Kalman Filter*, 2017.
- [14] Håvard Fjær, Lars Imsland, T. Johansen, T. Fossen, J. Kalkkuhl y A. Suissa, «Nonlinear vehicle side-slip estimation with friction adaptation.,» 2007.
- [15] E. L. J-H Yoon, *Estimation of vehicle sideslip angle and tire-road friction coefficient based on magnetometer with GPS*, 2015.
- [16] B. S. Wang Wei, *Vehicle Sideslip Angle Estimation Based on General Regression Neural Network*, 2016.
- [17] B. D. Boada, *Vehicle sideslip angle measurement based on sensor data fusion using an integrated ANFIS and an Unscented Kalman Filter algorithm*, 2016.

- [18] H. Sasaki, «Slip angle estimation device for a vehicle». Estado Unidos Patente US2020/0198646 A1, 25 Enero 2020.
- [19] J. Lu, «Vehicle Side Slip angle estimation using dynamic blending and considering vehicle attitude information.». Estados Unidos Patente US6,671,595 B2, 30 diciembre 2003.
- [20] P. P. N. V. Nguyen, «Observer, Estimation of Vehicle Dynamics States Using Luenberger,» *International Journal of Mechanical Engineering and Robotics Research*, vol. 8, nº 3, 2019.
- [21] L. Cardona, «Sistema de Navegación para vehículos no tripulados,» 2008.
- [22] L. Zadeh, *Redes Neuronales y Sistemas Difusos*, 2 Edición, 2001.
- [23] M. G. Daniel Chindamo, «Estimation of Vehicle Side-Slip Angle Using an Artificial Neural Network,» Brescia, Italy.
- [24] J. C. Fraile, *Control de la dinámica de vehículos eléctricos con motores en las ruedas*, Sevilla, 2019.
- [25] A. H. a. M. Simpson, «Estimation of vehicle sideslip angle and yaw rate,» *SAE World Congr., Detroit*, nº 2000-01-0696, 2000.
- [26] A. Nishio, K. Tozu, H. Yamaguchi, K. Asano y Y. Amano, «Development of vehicle stability control system based on vehicle side slip angle estimation,» *SAE*, nº 2001-01-0137, 2001.
- [27] A. Suissa, Z. Zomotor y F. Böttiger, «Method for determining variables characterizing vehicle handling,» *U.S. Patent*, nº 5557520, 1996.
- [28] G. Maudes, «Estimación del ángulo de deriva de un vehículo mediante redes neuronales,» 2013.
- [29] «Cálculo y análisis dinámico del automóvil,» de *Automóviles de turismo*, pp. 1-41.
- [30] G. M. COMENDADOR, *ESTIMACIÓN DEL*, Leganés, 2013.

ANEXOS

Parámetros del vehículo Class-B

```

clear all
%Datos de CarSim
load('C:\Users\Adrián\Documents\MIERA\TFM\Simulacion
observador\VariablesDataSim.mat');
Dt = data{1}.Values;
YR = data{2}.Values;
Al = data{3}.Values;
SA = data{4}.Values;

%Datos del Vehículo
Largo=2.8; %m
a=1.165; %m
Izz=996; %kgm2
Fy=5345.91;
Caf=(Fy*180)/(4*pi);
Car=Caf;
m = 1020+60+60; %kg
b=Largo-a;
Mz = 0.001;

%Velocidad del vehículo en la simulación 1
%V=120; %km/h
%Velocidad del vehículo en la simulación sinusoidal
%V=50; %km/h
%Velocidad del vehículo en la simulación circular
%V=40; %km/h

V=120*1000/3600;%m/s
Vx=V;%m/s

%Sistema del modelo de la bicicleta
A = [-(Caf+Car)/(m*Vx) ((-a*Caf+b*Car)/(m*Vx^2))-1; (-a*Caf+b*Car)/Izz (-
(a^2*Caf+b^2*Car)/(Izz*Vx))];
B = [Caf/(m*Vx); a*Caf/Izz];
C=[0 1];
D=0;

%State-space to TF
SS=ss(A,B,C,D);
TF=tf(SS);
VP=eig(A);
%Observador de Luenberger
%Polos de la matriz L
[Ka,Se,ee] = lqry(SS,10,0.1,0);
L=[Ka(1) Ka(2)]';

%Filtro de Kalman
%Discretizar matrices
SS_K=ss(A,B,C,D);
SS_d=c2d(SS_K,0.01);
TF=tf(SS_d);
A_d= SS_d.A;
B_d= SS_d.B;
C_d=SS_d.C;

```

```

D_d=SS_d.D;

G= [0.01;0.3];
H=0.5;
Q=10;
R=0.1;
N=0;

sim('EstimacionSlipAngle.slx',8);

```

Matrices del vehículo class-B

- Giro circular a 40 km/h

$$A = \begin{bmatrix} -12.097 & -0.7443 \\ 36.1346 & -27.8882 \end{bmatrix} \quad (0-1)$$

$$B = \begin{bmatrix} 6.0454 \\ 89.5676 \end{bmatrix}$$

$$C = [0 \quad 1]$$

$$D = 0$$

$$L = \begin{bmatrix} 0.3858 \\ 9.6939 \end{bmatrix}$$

Matrices discretas para Filtro de Kalman:

$$A = \begin{bmatrix} 0.885 & -0.0061 \\ 0.2961 & 0.7556 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.0540 \\ 0.7909 \end{bmatrix}$$

$$C = [0 \quad 1]$$

$$D = 0$$

- Doble cambio de carril a 120 km/h

$$A = \begin{bmatrix} -4.0302 & -0.9716 \\ 36.1346 & -9.2961 \end{bmatrix} \quad (0-2)$$

$$B = \begin{bmatrix} 2.0151 \\ 89.5676 \end{bmatrix}$$

$$C = [0 \quad 1]$$

$$D = 0$$

$$L = \begin{bmatrix} 0.3960 \\ 9.8964 \end{bmatrix}$$

Matrices discretas para Filtro de Kalman:

$$A = \begin{bmatrix} 0.9588 & -0.0091 \\ 0.3379 & 0.9096 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.0156 \\ 0.8583 \end{bmatrix}$$

$$C = [0 \quad 1]$$

$$D = 0$$

- Movimiento sinusoidal a 50 km/h

$$A = \begin{bmatrix} -9.6726 & -0.8363 \\ 36.1346 & -22.3106 \end{bmatrix} \quad (0-3)$$

$$B = \begin{bmatrix} 4.8363 \\ 89.5676 \end{bmatrix}$$

$$C = [0 \quad 1]$$

$$D = 0$$

$$L = \begin{bmatrix} 0.3890 \\ 9.7542 \end{bmatrix}$$

Matrices discretas para Filtro de Kalman:

$$A = \begin{bmatrix} 0.9065 & -0.0071 \\ 0.3080 & 0.7988 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.0427 \\ 0.8103 \end{bmatrix}$$

$$C = [0 \quad 1]$$

$$D = 0$$

Presentación de resultados

```
sim('EstimacionSlipAngle.slx',8);

figure
plot(BetaLineal)
hold on
plot(BetaCarSim)
title('Slip Angle')
ylabel('Beta [°]')
xlabel('tiempo [s]')
hold off
legend('Beta-Lineal','Beta-CarSim')

figure(2)
plot(YRLineal)
hold on
plot(YRCarSim)
title('Ang. de guiñada')
ylabel('YawRate [°]')
xlabel('tiempo [s]')
hold off
legend('YR-Lineal','YR-CarSim')

figure(3)
plot(DtCarSim)
title('Ang. de giro')
ylabel('Delta [°]')
xlabel('tiempo [s]')
%axis([0 inf 0 inf])

figure(4)
plot(AyLineal)
hold on
plot(AlCarSim)
title('Aceleración lateral')
ylabel('Ay [m/s²]')
```

```

xlabel('tiempo [s]')
hold off
legend('Ay-Lineal', 'Ay-CarSim')
%axis([0 inf 0 inf])

figure(5)
plot(tiempo,error)
title('Error Lineal-CarSim')
ylabel('error')
xlabel('tiempo [s]')

%Observador
figure (6)
plot(BetaObs)
hold on
plot(BetaCarSim)
title('Slip Angle')
ylabel('Beta [°]')
xlabel('tiempo [s]')
hold off
legend('Beta-Observador', 'Beta-CarSim')

figure(7)
plot(YRObs)
hold on
plot(YRCarSim)
title('Ang. de guiñada')
ylabel('YawRate [°]')
xlabel('tiempo [s]')
hold off
legend('YR-Observador', 'YR-CarSim')

figure(8)
plot(tiempo,error1)
title('Error Observador-CarSim')
ylabel('error1')
xlabel('tiempo [s]')

%Kalman
figure (9)
plot(BetaKalman)
hold on
plot(BetaCarSim)
title('Slip Angle')
ylabel('Beta [°]')
xlabel('tiempo [s]')
hold off
legend('Beta-Kalman', 'Beta-CarSim')

figure(10)
plot(YRKalman)
hold on
plot(YRCarSim)
title('Ang. de guiñada')
ylabel('YawRate [°]')
xlabel('tiempo [s]')
hold off
legend('YR-Kalman', 'YR-CarSim')

figure(11)
plot(error2)
title('Error Kalman-CarSim')
ylabel('error')

```

```
xlabel('tiempo [s]')
```

Parámetros del vehículo FOX

```
clear all

%Datos del Vehículo FOX
%Largo=3.2; %m
a=1.3; %m
Izz=1047.5; %kgm2
Fy=5345.91;
Caf=47000;
Car=53000;
m = 400.24; %kg
b=1.22;
Mz = 0.001;

%Velocidad del vehículo en la simulación
%V=120; %km/h
%V=10*1000/3600;%m/s
Vx=5;%m/s

load('C:\Users\Adrián\Documents\MIERA\TFM\Pruebas Pista\26_06\pruebas.mat')
load('C:\Users\Adrián\Documents\MIERA\TFM\Pruebas Pista\01_07\pruebas1.mat')

%t00 = 0:0.025:149.975;
t00 = 0:0.01:59.99;
t00 = t00';
Delta0 = timeseries(prueba.FOX_DATA(:,20)', t00);
Vx0 = timeseries(prueba.FOX_DATA(:,24), t00);
YawRate0 = timeseries(prueba.FOX_DATA(:,4), t00);
Beta0 = timeseries(prueba.FOX_DATA(:,3), t00);
accel00 = timeseries(prueba.FOX_DATA(:,29), t00);

%Sistema del modelo de la bicicleta
A = [-(Caf+Car)/(m*Vx) ((-a*Caf+b*Car)/(m*Vx^2))-1; (-a*Caf+b*Car)/Izz (-
(a^2*Caf+b^2*Car)/(Izz*Vx))];
B = [Caf/(m*Vx); a*Caf/Izz];
C=[0 1];
D=0;

%State-space to TF
SS=ss(A,B,C,D);
TF=tf(SS);
%Observador de Luenberger
%Polos de la matriz L
[Ka,Se,ee] = lqry(SS,10,0.1,0);
L=[Ka(1) Ka(2)]';

%Sistema del modelo de la bicicleta
A = [-(Caf+Car)/(m*Vx) ((-a*Caf+b*Car)/(m*Vx^2))-1; (-a*Caf+b*Car)/Izz (-
(a^2*Caf+b^2*Car)/(Izz*Vx))];
B = [Caf/(m*Vx); a*Caf/Izz];
C=[0 1];
D=0;

%Filtro de Kalman
%Discretizar matrices
SS_K=ss(A,B,C,D);
SS_d=c2d(SS_K,0.01);
TF=tf(SS_d);
A_d= SS_d.A;
```

```

B_d= SS_d.B;
C_d=SS_d.C;
D_d=SS_d.D;

G= [0.01;0.3];
H=0.5;
Q=10;
R=0.1;
N=0;

sim('EstimacionSlipAngleFOX.slx',50);

```

Matrices del vehículo FOX

$$A = \begin{bmatrix} -49.97 & -0.6442 \\ 3.3986 & -30.2272 \end{bmatrix} \quad (0-4)$$

$$B = \begin{bmatrix} 6.0454 \\ 89.5676 \end{bmatrix}$$

$$C = [0 \quad 1]$$

$$D = 0$$

$$L = \begin{bmatrix} 0.0509 \\ 9.4962 \end{bmatrix}$$

Matrices discretas para Filtro de Kalman:

$$A = \begin{bmatrix} 0.6066 & -0.0043 \\ 0.0228 & 0.7391 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.1834 \\ 0.5064 \end{bmatrix}$$

$$C = [0 \quad 1]$$

$$D = 0$$

Presentación de resultados

```

sim('EstimacionSlipAngleFOX.slx',50);

figure
plot(BetaLineal)
hold on
plot(Beta0)
title('Slip Angle')
ylabel('Beta [°]')
xlabel('tiempo [s]')
hold off
legend('Beta-Lineal','Beta-FOX')
axis([0 50 -inf inf])

figure(2)
plot(YRLineal)
hold on

```

```

plot(YawRate0)
title('Ang. de guiñada')
ylabel('YawRate [°]')
xlabel('tiempo [s]')
hold off
legend('YR-Lineal','YR-FOX')
axis([0 50 -inf inf])

figure(3)
plot(Delta0)
title('Ang. de giro')
ylabel('Delta [°]')
xlabel('tiempo [s]')
%axis([0 50 -inf inf])
axis([0 50 -inf inf])

figure(4)
plot(AyLineal)
hold on
plot(accel00)
title('Aceleración lateral')
ylabel('Ay [m/s2]')
xlabel('tiempo [s]')
hold off
legend('Ay-Lineal','Ay-FOX')
%axis([0 inf 0 inf])
axis([0 50 -inf inf])

figure(5)
plot(error)
title('Error Lineal-FOX')
ylabel('error')
xlabel('tiempo [s]')

%Observador
figure(6)
plot(BetaObs)
hold on
plot(Beta0)
title('Slip Angle')
ylabel('Beta [°]')
xlabel('tiempo [s]')
hold off
legend('Beta-Observador','Beta-FOX')
axis([0 50 -inf inf])

figure(7)
plot(YRObs)
hold on
plot(YawRate0)
title('Ang. de guiñada')
ylabel('YawRate [°]')
xlabel('tiempo [s]')
hold off
legend('YR-Observador','YR-CarFOX')
axis([0 50 -inf inf])

figure(12)
plot(error1)
title('Error Observador-FOX')
ylabel('error')
xlabel('tiempo [s]')

figure(8)
plot(Vx0)

```

```

title('Velocidad Longitudinal')
ylabel('velocidad [m/s]')
xlabel('tiempo [s]')
axis([0 50 -inf inf])

%Kalman
figure (9)
plot(BetaKalman)
hold on
plot(Beta0)
title('Slip Angle')
ylabel('Beta [°]')
xlabel('tiempo [s]')
hold off
legend('Beta- Kalman','Beta-FOX')
axis([0 50 -inf inf])

figure(10)
plot(YRKalman)
hold on
plot(YawRate0)
title('Ang. de guiñada')
ylabel('YawRate [°]')
xlabel('tiempo [s]')
hold off
legend('YR- Kalman','YR-FOX')
axis([0 50 -inf inf])

figure(11)
plot(error2)
title('Error Kalman-FOX')
ylabel('error')
xlabel('tiempo [s]')

```

Red Neuronal

```

import torch.nn as nn
import torch.nn.functional as F
import torch
import torch.optim as optim
import numpy as np
import pandas
import matplotlib.pyplot as plt

class NeuralNetwork(nn.Module):

    def __init__(self,num_inputs, num_outputs,lr):
        super(NeuralNetwork, self).__init__()

        self.input = nn.Linear(num_inputs,10)

        self.hidden = nn.Linear(10,10)
        self.hidden2 = nn.Linear(10,10)
        self.hidden3 = nn.Linear(10,10)

```

```

        self.output = nn.Linear(10, num_outputs)

        self.optimizer = optim.Adam(self.parameters(), lr=lr)

        self.loss = nn.MSELoss()

    def forward(self, x):

        x = self.input(x)
        x = self.hidden(x)
        x = self.hidden2(x)
        x = self.hidden3(x)
        out = self.output(x)

        return out

def main():

    # Creamos el objeto red neuronal #
    neural_network = NeuralNetwork(4,1, lr = 1e-4)

    data = pandas.read_csv('todos.csv')
    features = ['AyG_S1', 'Steer_SW', 'Vx', 'AVz']

    # Entrenamiento #

    num_of_batchs = 10000
    batch_size = 20

    loss_buff = []

    for i in range(num_of_batchs):

        neural_network.zero_grad()

        data_input = data.sample(batch_size)

        inputs = torch.tensor(data_input[features].to_numpy()).float()

        y_pred = neural_network(inputs)

        y_true = torch.tensor(data_input['Beta'].to_numpy().reshape(-
1,1)).float().detach()

        loss = neural_network.loss(y_pred, y_true)

        loss_buff.append(loss)

```

```
loss.backward()

neural_network.optimizer.step()

print('Done!')

plt.figure()
plt.plot(loss_buff, 'b')
plt.grid()
plt.xlabel('Training step')
plt.ylabel('Loss')
plt.show()

# Validacion sobre el train set (trivial) #

inputs = torch.tensor([data[features].to_numpy()]).float()
beta_estimada = neural_network(inputs)
beta_real = data['Beta'].to_numpy()

plt.figure()
plt.plot(beta_estimada.data.numpy()[0], 'r')
plt.plot(beta_real, 'b')
plt.grid()
plt.legend(['Beta estimada', 'Beta real'])
plt.show()

validate_data = pandas.read_csv('FOX1.csv')
inputs = torch.tensor([validate_data[features].to_numpy()]).float()
beta_estimada = neural_network(inputs)
beta_real = validate_data['Beta'].to_numpy()

plt.figure()
plt.plot(beta_estimada.data.numpy()[0], 'r')
plt.plot(beta_real, 'b')
plt.grid()
plt.legend(['Beta estimada', 'Beta real'])
plt.show()

if __name__ == '__main__':

    main()
```