

UNIVERSIDAD AUTONOMA DE MADRID

ESCUELA POLITECNICA SUPERIOR



PROYECTO FIN DE CARRERA

**DESARROLLO DE UN SISTEMA DE
CONTROL Y PRUEBAS PARA UNA
PLATAFORMA DE MOVIMIENTO CON 6
GRADOS DE LIBERTAD**

Sergio Orzáez Sánchez

NOVIEMBRE 2010

TÍTULO: Desarrollo de un sistema de control y pruebas para una plataforma de movimiento con 6 grados de libertad

AUTOR: Sergio Orzáez Sánchez
TUTOR: Gabriel Rouzaut Subirá (INDRA)
PONENTE: Guillermo González de Rivera

**Dpto. de Tecnología Electrónica
y de las comunicaciones**

**Escuela Politécnica Superior
Universidad Autónoma de Madrid**



AGRADECIMIENTOS

Cuando llega el momento de escribir estas líneas irremediablemente hay que echar la vista atrás y recordar lo que ha acontecido hasta llegar a este punto. Siempre y en lugar privilegiado está mi familia; mis padres, hermano, abuelos, tíos, ellos saben de lo importante de su presencia y apoyo cada día. Son parte de mi proyecto de vida.

Alex, Alberto, Manu, Guille, Javi, Antonio y Bea, los amigos que sienten como propios tus éxitos y te animan a lograrlos, gracias por estar siempre ahí.

Isa, Jesús, Eva, Vir, Cris, Talón, Marina, Álvaro, Marta, Sara, Marcos, codo con codo a lo largo de estos años, compañeros y sobretodo ahora, amigos.

Por supuesto agradecer a Juan Carlos Arenal e Ismael Ayala la oportunidad que me brindaron de realizar este proyecto, me llevo un gran recuerdo de los meses de trabajo en San Fernando. Una gran experiencia. Desear la mejor de las suertes al departamento y sus futuros proyectos. Y a Guillermo Glez. de Rivera por el interés mostrado por mi trabajo y por guiarme como ponente de este pfc.

A lo largo de estos años la Escuela ha sido el centro de grandes momentos y otros no tan memorables, pero lo que si tengo claro es que es el pilar en el que se edificará mi carrera profesional. Gracias a los que lo han hecho posible.

RESUMEN

Los simuladores reproducen comportamientos, es decir, sensaciones que en realidad no están sucediendo. En este proyecto está centrado en los simuladores de vehículos terrestres. La sensación visual se obtiene con proyecciones en pantallas de la realidad virtual del entorno, además de que el usuario se maneja dentro de una réplica del vehículo. Para calcar sensaciones físicas se usan plataformas hidráulicas que son dirigidas por un ordenador. Estos hacen uso de algoritmos que consiguen replicar las sensaciones de aceleración y velocidad en el conductor.

Lograr esa experiencia de conducción de un vehículo terrestre a través de esos algoritmos es el objetivo de este proyecto. Los datos de entrada son las aceleraciones lineales y los ángulos de movimientos a simular. Seis entradas, una por cada grado de libertad de la plataforma de movimiento. Se llevó a cabo un estudio de los distintos modelos matemáticos. El algoritmo de Washout es el que cumplía con los requisitos requeridos. Tras su análisis, estudio de las referencias, diseño de los filtros... se implementó el código de control de la plataforma.

El ajuste de las frecuencias y ganancias de los filtros requiere de una serie de pruebas que primeramente son llevadas a cabo en un PC con datos reales del simulador.

Finalmente, el proyecto estará en condiciones para realizar ensayos en un simulador de conducción en las instalaciones de Indra. Para facilitar las mismas se creó un programa de entorno visual donde variar los parámetros de configuración.

PALABRAS CLAVES

- Simulador de conducción
- Washout
- Filtro
- *Tilt-coordination*
- 6 grados de libertad
- Sistema vestibular

ABSTRACT

The simulators reproduce behaviours, in other words, sensations produced by something that is not really happening. This Project is focused on car simulators. The visual perceptions are obtained through projections on virtual reality screens. As a matter of fact, the user is sitting in a replica of a real car. To perceive the physical sensations hydraulic platforms, that are manipulated by a computer, are used. Making use of algorithms, computers can simulate acceleration and speed, and transmit this to the user.

The purpose of this project is to simulate the experience of driving a regular car through these algorithms. The data input are the lineal accelerations and the movement angles to be simulated. Six inputs, one for each freedom of movement degree of the platform.

A survey of the different mathematical models has taken place. The Washout algorithm complied with the demanded requirements. After its analysis, study of its references, design of the filters... the control code of the platform was implemented.

The frequency and gain filter tuning requires several tests which first take place on the PC with real data obtained from the simulator.

Finally, the project is ready to be used in a drive simulator to run tests at Indra facilities. In order to make these tests easier, a program of visual environment was created where the configuration parameters could be changed.

KEY WORDS

- Motion simulator
- Washout
- Filter
- *Tilt-coordination*
- 6 DoF
- Vestibular system

ÍNDICE DE CONTENIDOS

AGRADECIMIENTOS	3
RESUMEN	4
PALABRAS CLAVES.....	4
ABSTRACT	5
KEY WORDS.....	5
ÍNDICE DE CONTENIDOS.....	6
Índice de tablas.....	7
Índice de figuras.....	7
1. INTRODUCCIÓN.....	9
1.1. Motivación.....	9
1.2. Objetivos.....	10
1.3. Planificación	10
1.4. Organización de la memoria.....	11
2. ESTADO DEL ARTE.....	13
2.1. Simuladores.....	13
2.1.1. Introducción	13
2.1.2. Arquitectura del simulador	16
2.1.3. Simulador de conducción del vehículo terrestre de Indra.....	18
2.2. El sistema Vestibular	21
2.3. El sistema de control	23
2.3.1. Introducción	23
2.3.2. El Sistema de proceso.....	24
2.3.3. Algoritmos de control del movimiento.....	24
2.4. La plataforma	28
2.5. Tecnologías a usar	31
3. DESARROLLO DEL SISTEMA DE CONTROL.....	33
3.1. Esquema	33
3.2. Planteamiento basado en un sistema discreto.....	36
3.2.1 Diseño de un filtro digital	37
3.2.2. Diseño de un módulo de convolución	40
3.2.3. Diseño de un módulo integrador.....	40
3.2.4. Conclusión.....	41
3.3. Planteamiento basado en teoría de control	41
3.3.1. Filtros propuestos por Grant & Reid	41
3.3.2. Tratamiento de la señal de entrada y salida.....	45
3.3.3. FFT	46
3.3.4. Diseño de un módulo integrador.....	46
3.4. Cálculo de la posición (traslación).....	47
3.5. Simulación de la gravedad (<i>tilt-coordination</i>).....	51
3.6. Cálculo de la orientación (rotación).....	53
4. DESARROLLO DEL SISTEMA DE PRUEBAS.....	55
4.1. Características	55

4.2. Diseño e implementación	56
4.2.1. Primera versión	56
4.2.2. Mejoras en el sistema de pruebas	60
5. INTEGRACIÓN Y PRUEBAS DE LOS SISTEMAS.....	64
5.1. Integración.....	64
5.2. Pruebas con la base de datos.....	65
5.2.1. Introducción	65
5.2.2. Pruebas del cálculo de la posición	68
5.2.3 Resultados de orientación	75
5.3. Pruebas en el simulador.....	77
6. CONCLUSIONES Y TRABAJO FUTURO	79
BIBLIOGRAFÍA	81
Anexo A: LÍMITES DE LA PLATAFORMA	82
Anexo B: CONSTANTE DE GRAVEDAD	83
Anexo C: ¿CÓMO INSTALAR QT?.....	84
Anexo D: CAN BUS	87
Anexo E: VÍDEOS	88
PRESUPUESTO	89
PLIEGO DE CONDICIONES.....	90

Índice de tablas

Tabla 1-1: Planificación inicial del trabajo	11
Tabla 2-1: Sistemas que cooperan en el mantenimiento del equilibrio	21
Tabla A-1: Tilt Rate	82
Tabla A-2: Límites de movimiento de la plataforma	82

Índice de figuras

Figura 1-1: Modelos de vehículos a simular	9
Figura 2-1: Simulador ULTIMATE de Renault	16
Figura 2-2: Arquitectura de un simulador	17
Figura 2-3: Puesto de conducción simulador INDRA	19
Figura 2-4: Espejos retrovisores e interior	20
Figura 2-5: Reproducción del sistema visual	20
Figura 2-6: Oído interno	22
Figura 2-7: Sistema vestibular	22
Figura 2-8: Algoritmo de Washout	25
Figura 2-9: Washout adaptativo	26
Figura 2-10: Washout óptimo	27
Figura 2-11: Cabina sobre la plataforma de movimiento	29
Figura 2-12: Esquema plataforma Moog-6dof 2000e	30
Figura 2-13: Plataforma Moog-6dof 2000e	31
Figura 3-1: un modelo de Algoritmo de Washout	34
Figura 3-2: Señales y el algoritmo de Washout	35

Figura 3-3: Código de un filtro paso bajo ideal diseñado	38
Figura 3-4: Filtro paso bajo dominio frecuencial ($F_s=1800$; $F_c=500$; $N=30$) diseñado	38
Figura 3-5: Código del enventanado por Hamming	39
Figura 3-6: Filtro paso alto enventanado ($N=30$) diseñado	39
Figura 3-7: Código diseñado en MATLAB para hacer una convolución	40
Figura 3-8: Filtro paso alto diseñado, aceleración	43
Figura 3-9: Filtro paso bajo diseñado	44
Figura 3-10: Filtro paso alto diseñado, ángulos	45
Figura 3-11: Diagrama de bloques del cálculo de la posición de la plataforma	47
Figura 3-12: Gráfico coordenadas simulador terrestre	48
Figura 3-13: Punto de referencia	50
Figura 3-14: Diagrama <i>Tilt-coordination</i>	52
Figura 3-15: Fuerzas simuladas en el eje X	52
Figura 3-16: Diagrama de bloques del cálculo orientación de la plataforma	54
Figura 4-1: Interfaz Sistema de pruebas Washout	56
Figura 4-2: Ventana de ajustes parámetros generales	57
Figura 4-3: <i>Tool bar</i> de ajuste de frecuencias de corte y ganancias	57
Figura 4-4: Selección de ventanas a mostrar	58
Figura 4-5: Botones de inicio de ejecución y reseteo	58
Figura 4-6: Ventana de selección de la base de datos	59
Figura 4-7: Uso del zoom	60
Figura 4-8: Nueva barra de ajustes	61
Figura 4-9: Nueva ventana de parámetros generales	62
Figura 4-10: Ejemplo de funcionamiento	63
Figura 5-1: Extracto de la base de datos	65
Figura 5-2: Primeros datos	66
Figura 5-3: Ejemplo de funcionamiento	67
Figura 5-4: Grafica prueba ax (1)	68
Figura 5-5: Grafica prueba ax y salida pos X (1)	69
Figura 5-6: Grafica prueba ax (2)	70
Figura 5-7: Grafica prueba ax y salida pos X (2)	70
Figura 5-8: Grafica prueba ax (3)	71
Figura 5-9: Grafica prueba ax y salida pos X (3)	71
Figura 5-10: Grafica prueba ax (4)	72
Figura 5-11: Grafica prueba ax y salida pos X (4)	72
Figura 5-12: Grafica análisis estabilidad	73
Figura 5-13: Grafica prueba ax	74
Figura 5-14: Gráfica salida pos Y	75
Figura 5-15: Grafica ángulo heave	75
Figura 5-16: Grafica ángulo pitch	76
Figura 5-15: Grafica ángulo roll	77
Figura 5-16: Simulador de conducción (II)	78
Figura C-1: carpeta Qt	84
Figura C-2: Visual Studio 2005 Command Prompt	85
Figura C-3: About Qt	86

1. INTRODUCCIÓN

1.1. Motivación

Los simuladores son herramientas de apoyo en el proceso de aprendizaje y perfeccionamiento de habilidades. Esto es debido a que reproducen comportamientos, es decir, sensaciones que en realidad no están sucediendo. Se comenzaron a desarrollar debido a que ciertas actividades son peligrosas de ensayar sin un entrenamiento previo o para probar situaciones que no pueden hacerse de forma segura en un vehículo real.

Como sucede en muchas ocasiones en el campo de la tecnología, la industria militar fue la pionera en desarrollar los primeros simuladores de aviación a mediados del siglo XX. Hoy en día la industria civil también desarrolla simuladores para otros campos. En este proyecto nos centraremos en el estudio de vehículos terrestres.



Figura 1-1: Modelos de vehículos a simular

¿Cómo se alcanza a simular la conducción de un vehículo? La sensación visual se logra con proyecciones en pantallas de la realidad virtual del entorno deseado, además de que el usuario se maneja dentro de una réplica del automóvil.

El simulador cuenta con una plataforma de movimiento que se caracteriza por sus seis grados de libertad, esto quiere decir que el movimiento se produce en un espacio tridimensional, la plataforma tiene la capacidad de moverse hacia delante/atrás, arriba/abajo, izquierda/derecha (traslación), a la vez que se combinan con la rotación de los tres ejes perpendiculares (*roll* (ψ), *pitch* (θ), *yaw* (ϕ)). Todos los movimientos son independientes unos de otros.

Desde el último cuarto del siglo XX se vienen desarrollando teorías para el control de plataformas de movimiento. Para calcar sensaciones físicas se usan plataformas hidráulicas que son dirigidas por un ordenador. Estos hacen uso de algoritmos cuyos resultados en los actuadores consiguen transmitir las sensaciones de aceleración y velocidad en el conductor.

1.2. Objetivos

El cliente, Indra Sistemas, necesitaba de un estudio de los algoritmos disponibles para el control de plataformas así como de una primera implementación. Esto servirá como base para futuras mejoras y avances. Para que esto sea posible se debía contar con un mecanismo que permitiese comprobar de forma cómoda el buen funcionamiento del algoritmo que rige el sistema de control, dando la opción de realizar ajustes sobre los parámetros del mismo.

El objetivo de la empresa en definitiva es el de lograr una correcta experiencia de conducción de un vehículo terrestre a través del algoritmo de control escogido. Es decir, se conseguirá un control efectivo de los actuadores de la plataforma.

Los algoritmos consisten en rutinas de filtrado para las que es necesario conocer detalles tales como conocer las particularidades de la plataforma, limitaciones físicas, posición del conductor; el sistema vestibular, para evitar mareos del conductor; y finalmente escoger convenientemente frecuencias de corte y ganancias de los filtros, para lograr el propósito del simulador y "engañar" a los sentidos.

La correcta combinación de estas variables requiere de pruebas para ajustar el modelo implementado lo mejor posible. Se programará un entorno de pruebas que facilite la realización de esta práctica. En una primera aproximación estos ensayos serán en un PC con datos reales del simulador para, a continuación, el cliente pueda estar en disposición de pasar a comprobar el comportamiento del software desarrollado en el simulador. En esta última fase se verificará que el conductor percibe las correctas sensaciones como resultado del manejo del simulador.

1.3. Planificación

Este proyecto se ha realizado en seis meses, duración que estipulaba el contrato. Durante este tiempo se han cumplido diferentes etapas de cara a lograr el objetivo planteado por el cliente:

- Documentación: Búsqueda de información, análisis y planificación.
- Desarrollo: Planteamiento del problema, implementación, pruebas y desarrollo de los sistemas.
- Validación y pruebas: Simulación en PC y ajustes.

A lo largo de todas las etapas se llevaba a cabo paralelamente un proceso de documentación.

Enero			Febrero				Marzo				Semana Santa	
Etapa	Documentación			Desarrollo								
semana	1	2	3	4	5	6	7	8	9	10		11
TAREAS	Información			Planteamiento Matemático					Pruebas de código			
	Planificación					Implementación						
			Decisión									
semana	1	2	3	4	5	6	7	8	9	10	11	12

Desarrollo							Validación y pruebas						
13	14	15	16	17	18	19	20	21	22	23	24	25	26
Estudio Qt		Entorno pruebas								Ajustes en simulador			
						Simulación PC banco datos							
13	14	15	16	17	18	19	20	21	22	23	24	25	26

Tabla 1-1: Planificación inicial del trabajo

1.4. Organización de la memoria

El cuerpo de la memoria está dividido en cuatro apartados claramente diferenciados:

ESTADO DEL ARTE: Se hace una breve referencia histórica de los simuladores, introduciendo conceptos generales sobre la percepción de la sensación de movimiento, una visión de los distintos algoritmos planteados para manejar una plataforma, se documenta la plataforma que usa el simulador de la empresa para terminar con los recursos que se van a usar para llevar a cabo el objetivo.

DESARROLLO DEL SISTEMA DE CONTROL: Se describe el algoritmo de Washout para conocer los módulos que hay que implementar. A continuación se hace un estudio sobre los filtros que se van a utilizar para finalmente poder desarrollar el algoritmo.

IMPLEMENTACIÓN DEL SISTEMA DE PRUEBAS: Detallaremos el desarrollo de la interfaz de usuario con las características propuestas por el cliente que nos permitirá ajustar en la siguiente etapa con mayor facilidad los diversos parámetros.

INTEGRACIÓN Y RESULTADOS: Una vez que los dos sistemas están completos se integran y a partir de una base de datos se estudian los resultados obtenidos. A partir de las diversas pruebas se logran unos ajustes generales de frecuencias y ganancias que permiten llevar el sistema de control creado al simulador para observar las reacciones del conductor.

En último lugar se incluyen las conclusiones extraídas del proyecto, las líneas propuestas como trabajo futuro, las referencias consultadas y anexos que completan la memoria.

2. ESTADO DEL ARTE

2.1. Simuladores

2.1.1. Introducción

Los simuladores de conducción tienen un amplio espectro de aplicaciones. Podemos encontrar simuladores en autoescuelas, centros de formación de conducción para la policía u otros cuerpos de seguridad, industria del automóvil, mundo del deporte del motor (F1, WRC...), servicios de urgencias (ambulancias...), etc.

Los simuladores de conducción son cada vez más usados para entrenar a conductores en todo el mundo. Los estudios han demostrado que estos simuladores son excelentes herramientas para una efectiva educación y un excelente campo de prácticas. Los simuladores pueden ser una solución económica en los casos donde la alternativa (coche, camión, autobús etc.) es mucho más cara. Una clara ventaja es la reducción de riesgos para el conductor cuando se enfrenta a un escenario complejo el cuál podría ser peligroso cuando se aplica en un ambiente no-virtual.

Otra ventaja es la posibilidad que se le presenta a un monitor-profesor de corregir a un alumno y mejorar sus habilidades a los mando de un vehículo. Existen varios tipos de simuladores de conducción que son usados como simuladores de entrenamiento como simuladores de máquinas de tren, de grúas, de camión, de coche de bomberos, de autobuses, de automóviles, etc.

La línea de investigación y mejora en la cuál el simulador actualmente inmerso, implica la adaptación inteligente de la velocidad, el efecto de los sistemas automáticos en la seguridad y la mejora de la comprensión del conductor de las señales de tráfico. En conjunto implementar en el simulador sistemas que forman parte de los nuevos vehículos.

Los usuarios de simuladores prefieren la configuración dinámica a la estática. Las sensaciones del simulador se aproximan más al objetivo de similitud con la realidad con el uso de plataformas de movimiento. En esta memoria siempre nos referiremos a sistemas con actuadores.

Para entender la evolución de los simuladores terrestres no hay que dejar de lado la simulación en el mundo de la aviación, pionera dentro de la simulación de movimiento. Fueron durante las dos grandes guerras que abatieron Europa en el siglo XX cuando se dieron los primeros pasos y aparecieron los primeros modelos en el campo militar. Hubo que esperar hasta los años 50 para que su uso se extendiera a las aerolíneas comerciales (Pan American) [1-4]

Los sistemas evolucionaban a la par que la informática y los sistemas digitales. No fue hasta 1962 cuando D. Stewart introduce un sistema que consiste en dos plataformas con 6 actuadores, los 6 DOF. La llamada plataforma de Stewart fue muy bien recibida. Su tamaño relativamente compacto, permitía relativamente grandes movimientos.

En los 70 ya se desarrollaron simuladores de aerolínea con actuadores hidráulicos controlando los 6 ejes de movimiento y que contaban con una cabina real. Las sensaciones de vuelo fueron mejoradas con los años no sólo en el aspecto del movimiento sino también en el visual perfeccionando el campo de visión y la calidad de la misma (HD)

Un simulador de conducción tiene que reproducir los factores y condiciones externas con las que el vehículo interactúa transmitiendo esa sensación a un conductor como si estuviese en su propio coche. Escenarios y eventos son implementados con suficiente realidad como para asegurar que los conductores se encuentran inmersos en esa experiencia.

Es por tanto que los simuladores facilitan una experiencia constructiva para los conductores noveles y permite aumentar la complejidad de los ejercicios para lograr una rápida madurez a los mandos del vehículo. Por otro lado los simuladores aportan a un conductor con experiencia la oportunidad de mejorar y probar distintas situaciones que se pueden dar en la realidad.

¿Qué aportan a las empresas e instituciones? Una oportunidad para educar a sus empleados en habilidades y destrezas que ayudarán a reducir los gastos de mantenimiento, mejorar la productividad y, lo más importante, para certificar la seguridad de sus acciones en todas las posibles situaciones.

Los fabricantes de automóviles también se benefician del uso de simuladores de conducción, especialmente en las primeras fases del desarrollo de un coche. Las maniobras como el cambio de carril, slalom y frenar girando son determinadas por una sensación subjetiva, la cuál no se consigue mediante modelos por ordenador.

El uso de sistemas de movimiento contribuye en la toma de decisiones iniciales en la fase de diseño y permite la integración de proveedores en esta fase. Varias evaluaciones subjetivas, como por ejemplo umbrales y efectos adversos, pueden ser sistemáticamente analizadas.

A lo largo de las últimas décadas del siglo pasado diversas marcas de automóviles (VW, Mazda, Renault, Nissan, Daimler-Benz, etc.), instituciones y empresas han desarrollado y posteriormente, evolucionado sus sistemas de simulación de conducción:

→ Volkswagen: El primer simulador de conducción fue construido por VW a principio de los 70. Se trataba de un coche con un sistema de 3 grados de libertad. Contaba con una pantalla plana delante del conductor que se sentaba en una plataforma.

→ VTI: El instituto sueco de tráfico es un activo jugador en el mundo de la simulación de conducción. Comenzaron en 1984 con el VTI-I, un simulador con 4 grados de libertad. La mitad de un coche sobre una plataforma de movimiento con una pantalla fija delante de él.

A finales de los 80 evolucionaron este simulador (VTI-II) a petición de una empresa de seguros sueca que requería un simulador de camión, el cuál tiene una mayor carga que el pasajero de un simulador de coche.

En 2004 el VTI-II crecía en tamaño y equipamiento con una tabla de vibración que permitía que el conductor experimentara baches (*high frequent road rumbles*)

→ Daimler-Benz: El primer simulador de conducción de *Daimler-Benz* fue desarrollado en 1985 y fue el primero que permitió ser conducido en 6 grados de libertad. Un hexápodo hidráulico, el cuál fue diseñado especialmente, realizaba el mayor movimiento desarrollado hasta la fecha.

Una cabina de camión o de coche es situada dentro de una bóveda en la cuál 6 proyectores CRT permitían tener un campo de visión de 180°.

En 1993 el simulador fue mejorado. La principal diferencia con el diseño del anterior era la renovación del sistema de movimiento lateral. Esta modificación hizo que mejorase considerablemente la calidad del simulador.

→ Mazda: La compañía japonesa introdujo su simulador de conducción (4DOF) en 1985 con el objetivo de reducir el número de accidentes de tráfico que habían experimentado un rápido crecimiento con el avance de las motorizaciones. El diseño es muy similar a los simuladores VTI.

→ Ford: La compañía americana introduce *Virttex* en 1994, una bóveda sobre un hexápodo hidráulico. El *Virttex* es renovado en 2001.

→ BMW: La marca con sede en Munich desarrolla un hexápodo hidráulico de 4 metros de altura con una pequeña pantalla montada sobre una plataforma de movimiento junto con un coche a tamaño real.

Este sistema fue completamente renovado en 2003. La plataforma está ahora manteniendo una bóveda y el conductor entra al simulador a través de una pasarela.

→ Nissan: En 1999 desarrolló su sistema de 6DOF.

→ NADS: En 2002 en la Universidad de Iowa se presentó el NADS-1 (*North American Driving Simulator*) En esa época era de lejos el simulador más avanzado. El porqué de esta afirmación reside en que se trataba de un simulador con 9DOF, es decir, esto consiste en una plataforma con dos movimientos (longitudinal y lateral) sobre la que viaja un hexápodo. En la parte superior encontramos una bóveda la cuál esconde un coche de tamaño completo.

→ Toyota: En 2007 y con el objetivo de realizar test que de otro modo en la realidad podrían ser peligrosos, Toyota desarrolló en Susono en un nuevo centro técnico su simulador de conducción. El diseño fue a grandes rasgos muy similar al del NADS.

→ Renault: En 2004 *Renault* sustituyó su sistema de simulación con 6DOF por uno con un hexápodo sobre una plataforma con dos grados de libertad (plano xy) llamado ULTIMATE. El diseño del sistema de movimiento corrió a cargo de *Bosch- Rexroth*. [5]



Figura 2-1: Simulador ULTIMATE de Renault

→INTA (instituto nacional de técnica aerospacial) Desarrollado en 2004 por *Lander Simulation & Training Solutions*. Se trataba de un simulador de autobús y camión llamado Tutor y con el objetivo de mejorar actitudes y habilidades al conducir.

2.1.2. Arquitectura del simulador

El simulador civil de conducción de Indra está organizado en torno a un ordenador central de simulación. El host cuenta con tres aplicaciones con tareas en la simulación:

- El ejecutivo: dinámica de movimiento, seguimiento del terreno, instrumentos, tráfico, peatones...
- Volante: lectura de la posición del volante, control de fuerza y velocidad del mismo.
- Plataforma: sistema de control de la plataforma de movimiento.

El ordenador central cuenta con tres redes Ethernet independientes:

- ⇒ Visual.- Conexión directa mediante cable cruzado con el master del visual. Pantallas y espejos (tft) Esta red cuenta con dos subredes más:
 - Conexión de los generadores de imágenes y el master visual
 - Conexión de los cinco proyectores y el master visual.
- ⇒ Cabina.- Esta red interconecta los siguientes elementos:
 - Sonidos
 - Sensores (pedales y otros controles)
 - Plataforma
 - Host

Dentro de la cabina contamos con dos redes CAN independientes con el host:

- Volante
- Cuadro de instrumentos

La cantidad de componentes electrónicos en los vehículos de motor crece constantemente. Dado que las técnicas de cableado convencional con rutas largas ya no son capaces de manejar la cantidad de datos que deben intercambiarse, se usan en su lugar CAN Bus. El término Controller Area Network (CAN) se refiere a un sistema de bus serial desarrollado especialmente para su uso en vehículos de motor.

El bus de datos CAN sirve como base para el intercambio de datos digitales entre sensores, actuadores y unidades de control, asegurando que varias unidades de control puedan procesar la información procedente de un sensor y controlar sus actuadores en consonancia. Además de las rutas de cableado cortas, el bus de datos CAN tiene la ventaja particular de que, si falla un componente, el resto del sistema continúa funcionando normalmente, reduciendo en gran medida el riesgo de un fallo total del sistema. Incluso los sistemas eléctricos más antiguos en el vehículo están pasando a ser controlados con creciente frecuencia mediante el bus de datos CAN para mayor seguridad y comodidad.

⇒ Puesto de instrucción.- Conexión directa mediante cable cruzado con la posición de instructor.

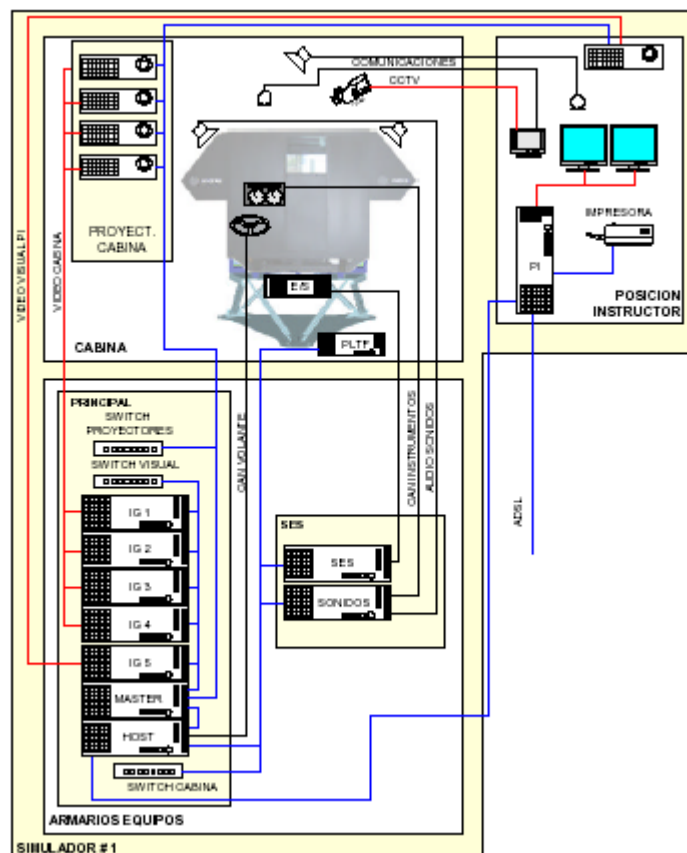


Figura 2-2: Arquitectura de un simulador

2.1.3. Simulador de conducción del vehículo terrestre de Indra

Indra Sistemas desarrolló el primer simulador terrestre con el propósito de servir de herramienta para la formación y evaluación de conductores, proporcionando un entorno dinámico y libre de riesgos, que reprodujese de forma controlada por un instructor las distintas situaciones reales que necesitan ser ejercitadas.

Las funciones que debía tener este simulador para dar solución a esas distintas situaciones son las siguientes:

1. Mejora del proceso de selección
2. Entrenamiento de los conductores de nuevo ingreso en las labores de conducción y manejo de equipos.
3. Cursos periódicos de actualización profesional.
4. Formación práctica para conductores pluriaccidentados

Para realizar el entrenamiento el simulador incorpora una cabina configurada con los elementos necesarios para la conducción de un turismo con caja de cambios manual de hasta 6 marchas adelante más marcha atrás.

La cabina incluye también un sistema de proyección de alta resolución en el que se presenta el entorno simulado, permitiendo a los alumnos:

- Potenciar la adquisición, desarrollo y mantenimiento de la destreza y habilidades necesarias para las tareas de conducción y manejo de los elementos reales del vehículo.
- Entrenar en el seguimiento de los procedimientos operacionales. Prevención de riesgos en condiciones normales de conducción. Práctica de conducción defensiva.
- Entrenar en la respuesta ante situaciones imprevistas y de emergencia. Prevención de riesgos en situaciones de emergencia en la conducción.
- Práctica de operaciones de control del vehículo en función de los distintos estados de la circulación o de la calzada y de sus variaciones con las condiciones atmosféricas.

Para realizar las labores de preparación, supervisión y evaluación de las sesiones de entrenamiento existe un puesto de instrucción.

El simulador de conducción está compuesto físicamente por los siguientes elementos:

- Cabina de conducción
- Puesto de Instructor
- Sistema de proceso (modelos matemáticos de simulación)
- Sistema Visual
- Sistema de Audio y Comunicaciones

Como puede observarse, los elementos más importantes para la conducción (volante, pedales, cambio, salpicadero, asiento, etc.) son partes reales de un turismo, en este caso de un Renault Scenic.



Figura 2-3: Puesto de conducción simulador INDRA

Se utiliza el salpicadero real del modelo elegido con sus correspondientes conexiones. Las distintas señales las genera el sistema de simulación en lugar del sistema central del automóvil real.

El modelo de vehículo simulado es un Renault Scenic, siendo posible la implementación de otros modelos utilizando los elementos de control del vehículo suministrados e incluiría todos los elementos necesarios para la conducción. El Simulador está diseñado para en caso necesario, posibilitar el entrenamiento en distintos modelos de vehículo en la cabina existente manteniendo los mismos mandos e indicadores. Para ello únicamente es necesario modificar la apariencia del vehículo en el visual y modificar los parámetros de funcionamiento del modelo (curvas de par, potencia, freno, respuesta de la dirección, sonidos, etc.).

Las pantallas del sistema visual se encuentran muy próximas a la posición de las superficies acristaladas del turismo, facilitando con ello la estimación de distancias a los obstáculos más cercanos al vehículo.

El objetivo del Sistema Visual es mostrar una reproducción realista de las escenas visuales observadas por el conductor. El sistema visual se compone del sistema de Generación de Imágenes (IG), basado en una red de PCs conectados en *cluster* y del sistema de presentación (pantallas, proyectores y espejos).

Los IGs contienen los datos relativos al entorno virtual, vehículo propio, vehículos del tráfico y humanos (peatones/pasajeros)

El sistema de presentación está formado en base a cuatro proyectores de alta resolución, tres pantallas de retroproyección y los espejos necesarios para dirigir la imagen desde los proyectores a las pantallas.



Figura 2-4: Espejos retrovisores e interior

- Dos canales frontales (parabrisas)
- Dos canales laterales (ventanillas).
- Un espejo interior (TFT)
- Dos espejos exteriores (TFTs sobre marco de espejo)

Todo el sistema de presentación visual (proyectores, espejos y pantallas) se encuentra embarcado sobre la plataforma de movimiento y se mueve solidario al puesto de conducción, por lo que el simulador mantiene campos de vista constantes e independientes de los movimientos de la plataforma.

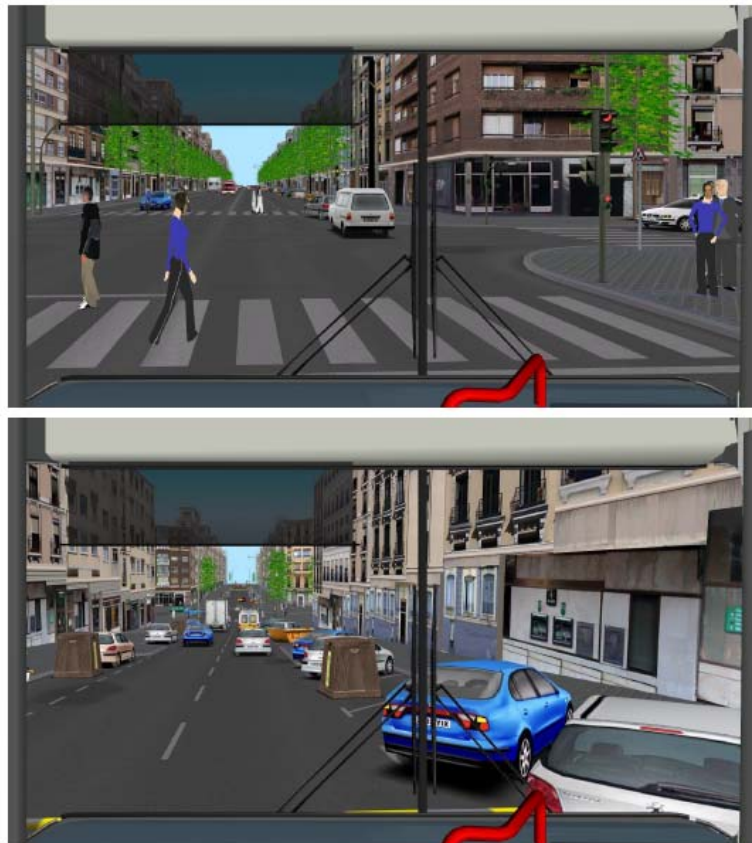


Figura 2-5: Reproducción del sistema visual

Las pantallas se encuentran situadas detrás de la parte de carrocería del vehículo instalada en el simulador, por lo que las zonas ocultas de la escena son las mismas que las obtenidas en el vehículo real, y de igual forma que en este pueden variarse moviendo levemente la cabeza.

La distancia de las pantallas a las superficies acristaladas correspondientes del vehículo (ventanillas izquierda y derecha y parabrisas) es reducida, facilitándose con ello la estimación de distancia a los objetos del escenario más próximos al vehículo.

Las tres pantallas están montadas y ajustadas de forma que horizontalmente se obtiene una imagen continua.

El sistema de presentación ofrece un campo de visión igual al que tiene el vehículo real, ocultándose el exceso por la parte de carrocería del vehículo instalada.

2.2. El sistema Vestibular

La simulación es definida como la imitación de algo real, conjunto de acontecimientos o procesos. La simulación de movimiento está estrechamente relacionada con la percepción. El sistema vestibular desempeña un papel fundamental en la orientación de los seres vivos gracias a los receptores que permiten la percepción de la aceleración lineal y angular.

Un conductor usa sus sentidos para obtener una base con la que tomar decisiones a partir de lo que percibe. El cuerpo humano tiene dos modos de percibir el movimiento: estímulos inerciales en el cuerpo y movimientos en el medio con respecto al cuerpo.

La simulación inercial proviene de la fuerza de la gravedad y las fuerzas externas y los momentos del cuerpo. En el sistema vestibular, localizado en el oído interior (izquierdo y derecho), está el sentido destacado que proporciona al sistema perceptivo la información sobre las aceleraciones lineales y angulares del cuerpo, relacionado con el equilibrio y el control espacial. El sentido de la percepción de la velocidad del cuerpo humano es logrado a través del sistema visual. El sentido de velocidad es también adquirido a través del sistema acústico.

La simulación de movimiento tiene por tanto que describir la presentación de información visual, acústica, vestibular y háptica para lograr su objetivo.

-Sistema vestibular (oído interno)
-Sistema visual.
-Receptores repartidos por todo el cuerpo y que informan de la posición de las articulaciones, músculos, etc.

Tabla 2-1: Sistemas que cooperan en el mantenimiento del equilibrio

El sistema vestibular [6] está contenido dentro del oído interno y formado por dos ensanchamientos: el utrículo y el sáculo, ambos informan de la posición de la cabeza en relación con el suelo. En estos ensanchamientos se encuentra un órgano receptor denominado mácula,

que está integrado por células receptoras ciliadas. Sobre los cilios de estas células sensoriales se hallan unos pequeños cristales de carbonato cálcico (CaCO_3) que reciben el nombre de otolitos.

Las células sensoriales de los otolitos son muy susceptibles a cambios de gravedad. Generan impulsos que llegan hasta el cerebelo informando de las posiciones de la cabeza respecto a la gravedad y a la aceleración de tipo lineal, lo que nos permite mantener el equilibrio a pesar de que realicemos desplazamientos, giros o aceleraciones.

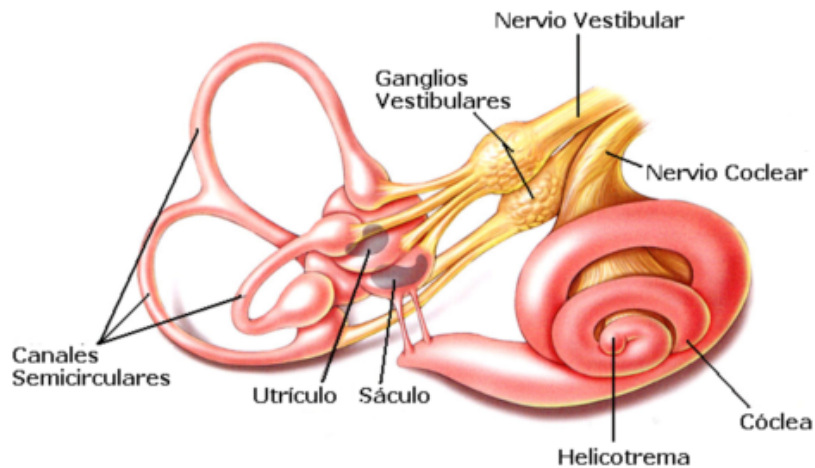


Figura 2-6: Oído interno

Las vías vestibulares comprenden las fibras que, partiendo de los conductos semicirculares y del vestíbulo, llegan hasta el tronco del cerebro y de aquí se dirigen a la corteza cerebral. Estas vías se inician en las Crestas Ampulares de los canales semicirculares y de las Máculas del utrículo y el sáculo.

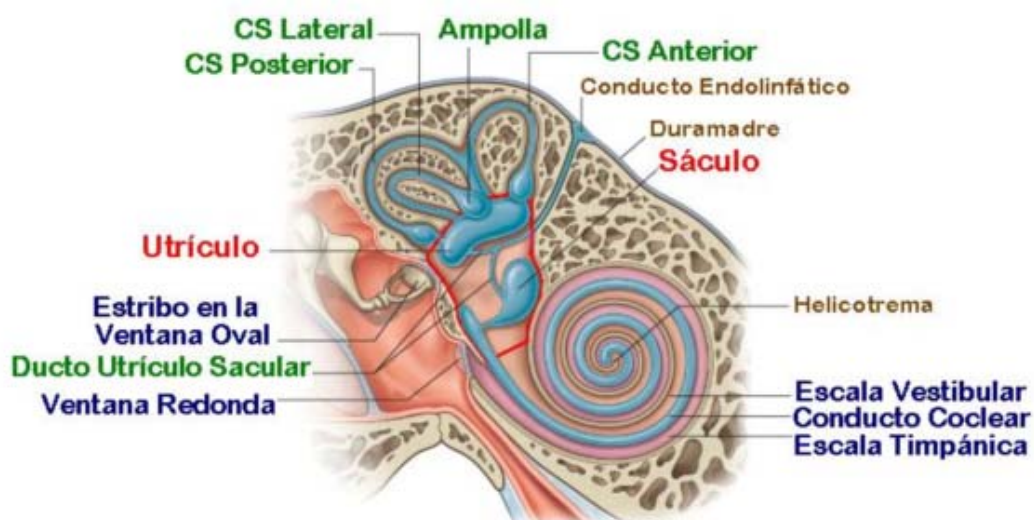


Figura 2-7: Sistema vestibular

Los conductos semicirculares son tres y están orientados en los tres planos del espacio. Presentan una dilatación en su parte inferior denominada "Ampolla", en el interior de la cual se encuentra un órgano del equilibrio, que recibe el nombre de cresta ampular. La cresta ampular está integrada por células sensoriales receptoras ciliadas, que están recubiertas por una membrana gelatinosa en forma de cúpula. Estas células descansan sobre otras de tipo conjuntivo, y conectadas con las neuronas que inician el nervio que conducirá la información hasta el interior del cerebro. Los conductos semicirculares informan de la aceleración angular (rotación).

El uso de un simulador puede provocar que los usuarios sufran algún tipo de indisposición o malestar. Los síntomas son el resultado de la diferencia en la latencia del visual y el sistema de movimiento además de las posibles falsas entradas. La latencia de movimiento debería ser menor o igual a la latencia visual y las falsas entradas deberían ser eliminadas.

El vértigo es un término amplio utilizado para explicar cómo nos sentimos cuando nuestra sensación de equilibrio no es la apropiada. Éste, junto al mareo, desequilibrio, desorientación, ruidos en los oídos y posiblemente náuseas, vómito, sudoración, entre otros, son el resultado de que el sistema vestibular (que proporciona información al cerebro sobre los cambios en movimiento) no está trabajando apropiadamente.

En un individuo con salud normal, los sistemas propioceptivos (pies, tobillos, articulaciones), vista (ojos) y el sistema vestibular trabajan juntos en armonía con el cerebro. Una persona con un trastorno del equilibrio, sin embargo, puede tener un problema en cualquiera de estos sistemas, o en sistemas múltiples.

En algunos individuos, uno o más de estos sistemas están fallando y la persona no se da cuenta de que está perdiendo su equilibrio. En otras personas, el cerebro se confunde y crea una sensación de caída cuando realmente se está en equilibrio. El riesgo a desarrollar uno o más de estos problemas se ha comprobado que es menor en usuarios que suelen trabajar con simuladores o videojuegos.

En resumen, un simulador de movimiento en su conjunto debe estar sincronizado y ser coherente para no provocar desajustes entre el sistema vestibular, visual y propioceptivo del individuo.

2.3. El sistema de control

2.3.1. Introducción

Cuando se desarrolla un producto de éxito es clave mantenerlo siempre actualizado para no perder su atractivo por tanto su competitividad en el mercado. Esa es la situación del simulador de conducción de Indra. Cuenta con un simulador con recorrido para evolucionar y mantenerse en condiciones de convencer a los posibles clientes.

Para desarrollar un producto hay que tener la posibilidad de poder intervenir en cuantos más procesos de su diseño y fabricación mejor. La empresa debe negociar por los mejores componentes de hardware como el *host*, pantallas, la plataforma o los módulos del automóvil con las correspondientes industrias. Ahora bien, la mejora de la respuesta de los módulos del simulador a los sensores, o el software, si que está controlado desde su concepción por la empresa y es ahí donde se tiene que realizar las evoluciones y mejoras. El visual, la generación y presentación de imágenes es una de las correspondientes áreas que se pueden ver envueltas en esta tarea, por ejemplo.

Si se tiene esta idea de negocio no hay ningún elemento en el simulador que pueda quedar rezagado. Dentro de la arquitectura del simulador una de las aplicaciones del *host* es el sistema de control de la plataforma que, antes de este proyecto, no podía seguir la política presentada ya que era un sistema cerrado, propietario, sin posibilidad de intervención.

Es por este motivo por el que se pretende crear una línea de desarrollo dentro del departamento para equiparar este sistema a la evolución del conjunto del producto.

2.3.2. El Sistema de proceso

Los elementos principales que "interactúan" en el simulador son el propio vehículo, el entorno y el conductor. La consecuencia de la acciones del conductor dentro del escenario que propone el simulador genera una serie de parámetros. Estos son procesados para obtener los datos que serán enviados a cada uno de los componentes del sistema.

El Sistema de Proceso es aquella parte del software que establece los ciclos de simulación, realizando el traspaso de información de los elementos de entrada de cabina a los modelos matemáticos, cálculo del nuevo estado de los modelos y traspaso de información de los modelos a los elementos de salida de la cabina y sistema visual.

Dentro del Sistema de Proceso se encuentran los modelos matemáticos que corresponden al software que implementa los diferentes sistemas de vehículo a simular. Este modelado incluye, entre otros, los sistemas de tracción, freno, dirección, eléctrico y modelo mecánico.

El sistema de control da respuesta gracias a la implementación de un algoritmo. Éste recibirá del sistema de proceso las entradas necesarias. Información relativa a la aceleración lineal y a los ángulos que ha sido obtenida por el sistema de proceso a partir de las acciones del usuario dentro del simulador.

2.3.3. Algoritmos de control del movimiento

El diseño del sistema de control depende de la arquitectura del simulador y del tipo de maniobras que estemos buscando reproducir. El número de grados de libertad de la plataforma así como el tipo de entradas que genere el sistema de proceso influyen en el diseño. La teoría de control que estudiamos fue propuesta para crear simuladores de vuelo pero es un punto de partida para lograr un efectivo control de un vehículo terrestre.

¿Cuáles son las características de un algoritmo de movimiento?

Los principales objetivos son optimizar la aceleración y mantener el sistema de movimiento dentro de sus limitaciones físicas.

En la bibliografía encontramos distintas soluciones para el control de una plataforma. Se trata de teorías que podríamos desarrollar e implementar a posteriori. [7-9]

→ WASHOUT CLÁSICO

Expresa la compensación en el dominio de la frecuencia: los comandos de la plataforma resultan de un filtrado paso alto de la aceleración del vehículo. Para eliminar la componente de baja frecuencia, las aceleraciones de alta frecuencia son devueltas mientras el desplazamiento de la plataforma se reduce. Fue propuesto por Reid y Nahon.

La selección de los parámetros de un filtro es un sacrificio entre la fidelidad y los límites físicos de la plataforma. El filtro es configurado para el peor caso, suponiendo que la aceleración del vehículo simulado es un paso de la señal de amplitud A_{max} . La pulsación $w1$ determina las componentes frecuenciales de la aceleración que serán rechazadas. Mientras tanto la pulsación wL controla la rapidez con la que la plataforma vuelve a su posición neutral.

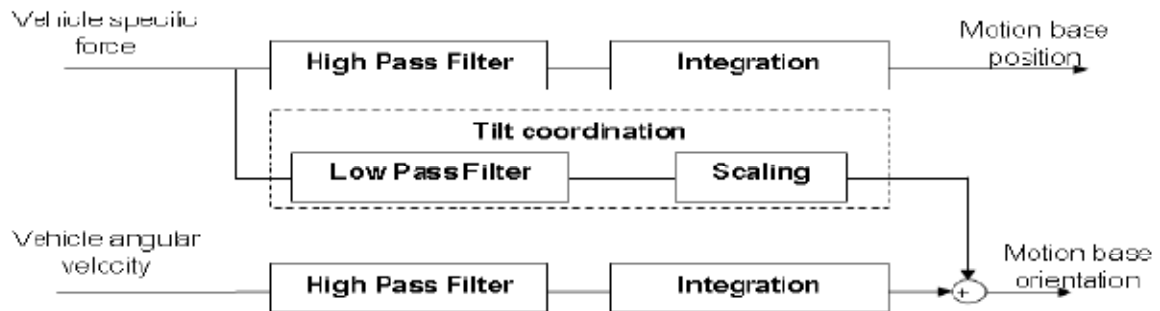


Figura 2-8: Algoritmo de Washout

Una vez que la forma del filtro clásico es definida, un proceso de experimentación prueba-error es realizado para establecer el valor de los diferentes parámetros del filtro. Un conjunto de parámetros son fijados, y al obtener resultados se corrige o se logra la solución del problema optimizado.

El inconveniente de este método es que algunas falsas entradas son introducidas por los abruptos cambios en las frenadas, debido a las características lineales del filtro paso alto. Estas pueden alterar la percepción del conductor.

Esto puede ser corregido introduciendo otros algoritmos y mecanismos que reduzcan ese contragolpe. Finalmente, los filtros son ajustados para el peor caso el cuál reduzca considerablemente el desplazamiento de la plataforma para otras maniobras.

Los parámetros óptimos obtenidos por este último método no son necesariamente óptimos para otras maniobras ya que configuramos para el peor caso para cada movimiento. El algoritmo clásico es el más sencillo pero tiene esta limitación del ajuste

El algoritmo clásico es el más apropiado para nuestro caso en término de percepción humana y diseño sencillo. El cliente optó por este desarrollo para reducir los plazos y tener una base con la que trabajar en un futuro e implementar mejoras. Todos los algoritmos desarrollados tras este toman el *washout* clásico como punto de partida, esto lo hace muy atractivo.

→ *ADAPTATIVE WASHOUT*

Basado en el dominio temporal y el de la frecuencia: su estructura deriva del algoritmo clásico, pero aquí, las ganancias y las frecuencias de corte de los filtros no son constantes. Se desarrolla un método para ajustar dinámica los parámetros para sacar el máximo partido a la plataforma.

A cada paso computacional, son elegidas para minimizar una función de coste que aúna el movimiento de la plataforma y los vacíos entre la aceleración del vehículo y la aceleración de la plataforma sobre una sesión de simulación infinita. Fue propuesto por Parrish.

Este algoritmo puede ser visto como uno clásico donde los parámetros son modificados y calculados a cada paso en tiempo de simulación. Varios esquemas fueron propuestos para mejorar la estabilidad del algoritmo.

Alguno de ellos basado en la minimización de una función de coste que contiene el error de aceleración y las restricciones de desplazamiento de la plataforma. La adaptación es llevada a cabo usando el método de descenso por gradiente para resolver la ecuación.

El hecho de que el algoritmo adaptativo permita el ajuste de los parámetros del filtro en "tiempo real" hace posible reducir algunas falsas entradas generadas por el algoritmo clásico.

Diversos estudios certifican que el retorno a la posición neutral es más lento que si se usa el algoritmo clásico, la percepción es más suave principalmente al principio de la aceleración. El inconveniente de este método es la dificultad de encontrar la función de coste con el peso más relevante, y los valores iniciales de los diferentes parámetros, que dan el mejor resultado mientras se asegura la estabilidad. Además de no tener en cuenta las sensaciones del movimiento.

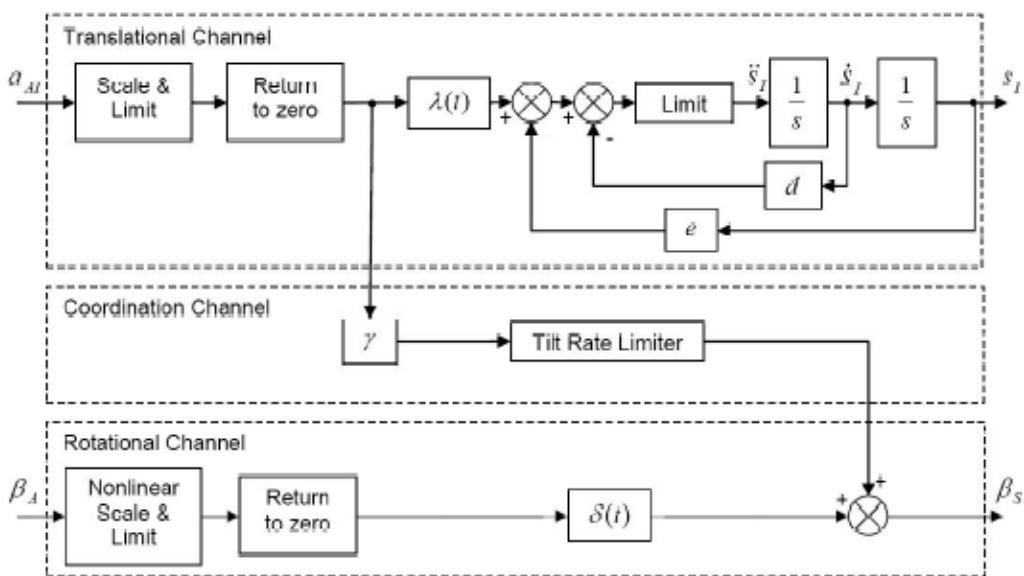


Figura 2-9 Washout adaptativo

El ajuste de las frecuencias del algoritmo clásico y el adaptativo para un simulador con un gran margen de movimiento puede indirectamente guiar hacia una gran diferencia en la aceleración física y/o en la percibida entre la configuración clásica y la adaptativa.

→ ALGORITMO ÓPTIMO

Fue Sivan (1982) quien estudió este algoritmo que es el resultado de una combinación de filtros paso alto y paso bajo lineales. Se establece un modelo que relaciona las aceleraciones y las rotaciones del vehículo con las que deberían ser aplicadas por la plataforma de tal modo que la diferencia con las sensaciones sentidas es mínima. Nos encontramos ante una derivación del algoritmo clásico, aquí también los comandos de la plataforma minimizan un coste global. Este coste corresponde al movimiento de la plataforma y el vacío entre aceleración del vehículo y la aceleración de la plataforma sobre una sesión infinita de simulación.

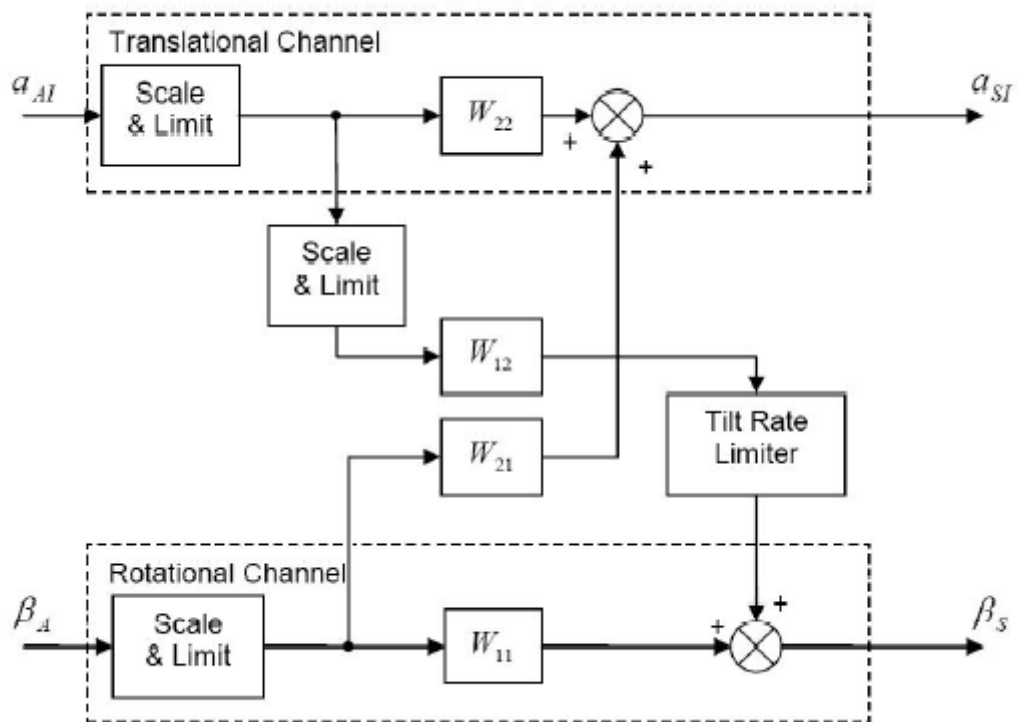


Figura 2-10: Washout óptimo

Usar el modelo óptimo requiere la manipulación de una serie de pesos (W_n) que frecuentemente no tienen una clara representación física lo cual puede convertirse en una tarea complicada. Posteriormente se presentó una técnica que minimizaba este problema recurriendo a la teoría de

patrones (algoritmos genéticos). El resultado final que se elige es el más apropiado al global de las situaciones pero no garantiza el mejor comportamiento en todas las situaciones.

En definitiva el algoritmo óptimo presenta diversas carencias. A pesar de que minimiza la sensación de error entre el conductor en el vehículo virtual y el de la plataforma de simulación. Los resultados obtenidos no son lo suficientemente buenos para lograr una excelente percepción.

→ OTRAS TEORÍAS

Se plantean otras opciones, como resolver compensando en el dominio del tiempo, es decir, calculando a cada paso el comando de la plataforma que minimiza la diferencia entre la aceleración del vehículo y la aceleración del sujeto simulado para limitaciones físicas explícitas y perceptivas. Se conoce como Algoritmo predictivo y su autor fue Dagdelen.

Recientemente, este método fue desarrollado para generar entradas de movimiento para el simulador de conducción de Renault.

Se están desarrollando también algoritmos que incorpora un modelo matemático del sistema vestibular humano, a partir de la teoría clásica de *washout*, restringiendo la sensación de error entre el vehículo simulado y el movimiento de la plataforma usa filtros de alto orden con métodos de control óptimos.

2.4. La plataforma

A la hora de desarrollar un sistema de simulación es importante recoger las opiniones y sugerencias de los usuarios para lograr un mejor producto. Estos confirman que el uso de plataformas hidráulicas añade una percepción del movimiento que mejora las cualidades del simulador en cuestión.

Lograr el mayor realismo posible es el objetivo final para ello se dispone de una cabina de conducción que es una reproducción a escala natural del puesto de conducción de un turismo y contiene toda la instrumentación real necesaria para que el usuario pueda realizar la conducción del vehículo.

La cabina completa incluyendo el sistema de presentación visual se encuentra montada sobre una plataforma de movimiento de seis grados de libertad.



Figura 2-11: Cabina sobre la plataforma de movimiento

La plataforma de movimiento (junto con el sistema visual) permite simular las sensaciones que se experimentan en un vehículo en movimiento. Con este sistema se reproducen los efectos de aceleración y frenada, las aceleraciones laterales debidas a los giros, las inclinaciones debidas a pendientes en la vía y el efecto que las irregularidades del firme como bordillos, resaltes y badenes tienen sobre el vehículo.

La plataforma de movimiento consta de dos estructuras unidas a través de seis actuadores. La inferior permanece fija en su posición en el suelo mientras que la superior varía su posición por medio de los actuadores. Encima de esta segunda estructura descansa la cabina o réplica del vehículo.

El movimiento de la plataforma (*Moog – 6dof*) se genera calculando la posición instantánea de cada uno de los actuadores a partir de las aceleraciones y ángulos obtenidos del modelo dinámico en cada ciclo de simulación. Es decir, las coordenadas de salida de nuestro algoritmo responden a la relación entre el movimiento de las articulaciones de la plataforma y el movimiento resultado de un sólido rígido. A tener en cuenta que no es necesario transformar las señales de salida del sistema de control porque el propio software de la plataforma en cuestión las transforma en coordenadas de posicionamiento de sus actuadores.

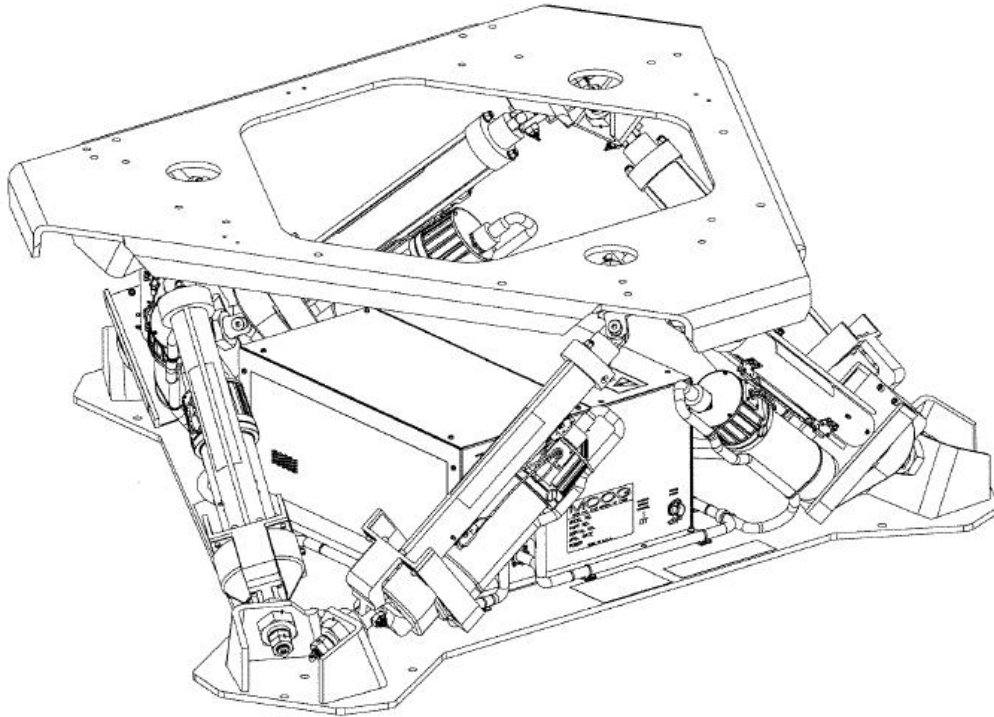


Figura 2-12: Esquema plataforma Moog-6dof 2000e

El hexápodo como el de la imagen permite obtener seis grados de libertad, referidos al movimiento de un sólido rígido (un vehículo en nuestro caso) en las tres dimensiones del espacio. Pasamos a analizar los ejes del movimiento:

→ Traslación:

X: Movimiento longitudinal, hacia delante o atrás.

Y: Movimiento lateral, izquierda o derecha.

Z: Movimiento vertical, arriba o abajo.

→ Rotación:

Heave/Yaw: Giro sobre el eje X.

Pitch: Inclinación. Giro sobre el eje Y

Roll: Rotación sobre el eje. Elevación lateral, curvar para compensar el efecto de la fuerza centrífuga durante un giro. En el caso de un coche lo experimentamos al subir un bordillo o giro a gran velocidad. Giro sobre el eje Z.



Figura 2-13: Plataforma Moog-6dof 2000e

Las salidas del algoritmo de control del movimiento no dependen sólo del ajuste de parámetros y de las tareas de conducción (entradas) sino también del espacio de trabajo de la plataforma desde el que se tiene que mantener al sistema de movimiento dentro sus límites.

La plataforma que utiliza el simulador está constituida por actuadores que cuentan con servomotores. Un servomotor es un dispositivo que tiene la capacidad de ubicarse en cualquier posición dentro de su rango de operación, y de mantenerse estable en dicha posición. Está formado por un motor de corriente continua, una caja reductora y un circuito de control.

2.5. Tecnologías a usar

El desarrollo del sistema de control y el de pruebas son programados en *c* y *c++*. Como editores de estos lenguajes se hace uso de la herramienta *Microsoft Visual Studio* para el sistema de control y de *Qt creador* para el sistema de pruebas. El código del sistema de pruebas está escrito usando *Qt*.

Qt es una biblioteca multiplataforma (Windows, Mac, Linux...) para desarrollar interfaces gráficas de usuario y también para el desarrollo de programas basado en *C++*. Incluye una librería, herramientas de desarrollo integradas y un IDE multiplataforma. Usando *Qt*, puedes escribir aplicaciones web una vez y luego hacer uso de ellas a través de escritorios y sistemas operativos

embebidos sin necesidad de reescribir el código fuente. Tan sólo habría que compilarlo en la plataforma elegida. (Ver anexo B)

Previo a la parte de implementación del código definitivo se hace uso de *Matlab* como banco de pruebas de las distintas partes del proyecto, particularmente el sistema de control.

Durante la etapa de estudio se utiliza Microsoft Visio para crear esquemas y diagramas de bloques para plantear el algoritmo.

Se hará uso del sistema de pruebas junto a una base de datos en un *PC*. Tras estas pruebas y ajustes se podrá llevar a cabo la prueba final del sistema de control en el simulador de conducción con plataforma *Moog*.

3. DESARROLLO DEL SISTEMA DE CONTROL

3.1. Esquema

El punto de partida que se propuso para el desarrollo del sistema fue el de la teoría clásica *washout* ya que posibilita que sea mejorado aplicando a partir de ella diferentes caminos de evolución sin cerrar puertas. Ya que en un futuro y, tras este proyecto, el departamento será quien determine hacia donde se dirige el desarrollo del sistema de control de la plataforma de movimiento. [10-11]

Cuando se juega a videojuegos se tiende a sentir como si se formase parte de ese mundo virtual con el que están interactuando. Perciben el movimiento a través de la pantalla. Entonces ¿por qué existe la necesidad de añadir movimiento? Un conductor debería sentir la misma aceleración que la que está percibiendo visualmente, este hace que la experiencia sea más enriquecedora pero sobretodo más real.

Como hemos visto en apartados anteriores para simular el movimiento, de modo físico, es necesario el uso de una plataforma de movimiento. Esta plataforma realizará la acción correspondiente a los datos que reciba. Estos han tenido que ser tratados con anterioridad para asegurarnos que están dentro de los límites físicos de la misma así como no sólo activan un cierto movimiento sino que este sirve para "traicionar" a los sentidos del usuario.

Aunque en otras plataformas sea necesario implementar una interfaz que interprete los datos que le pasemos, en este caso, por el modelo con el que se cuenta, no es necesario ya que está integrado.

Llegado a este punto se tiene claro que hay que diseñar y desarrollar posteriormente un algoritmo de control. El escogido para manejar el sistema de movimiento es el principio del algoritmo clásico de *Washout* que se puede ver en la figura 3-1.

En un primer momento se llamó "Washout" porque la plataforma es movida linealmente en la misma dirección de aceleración que reproduce las aceleraciones de tránsito longitudinal y laterales y vuelve a su posición neutral. Este último movimiento debería no ser detectado por el conductor (con respecto a la sensación de umbral) para no ser interpretado como una falsa entrada.

Sólo las componentes de alta frecuencia de las aceleraciones de traslación y rotación son reproducidas por el simulador. Esto requiere mantener al simulador de movimiento dentro de las características del sistema. El sistema visual proveerá la información de baja frecuencia.

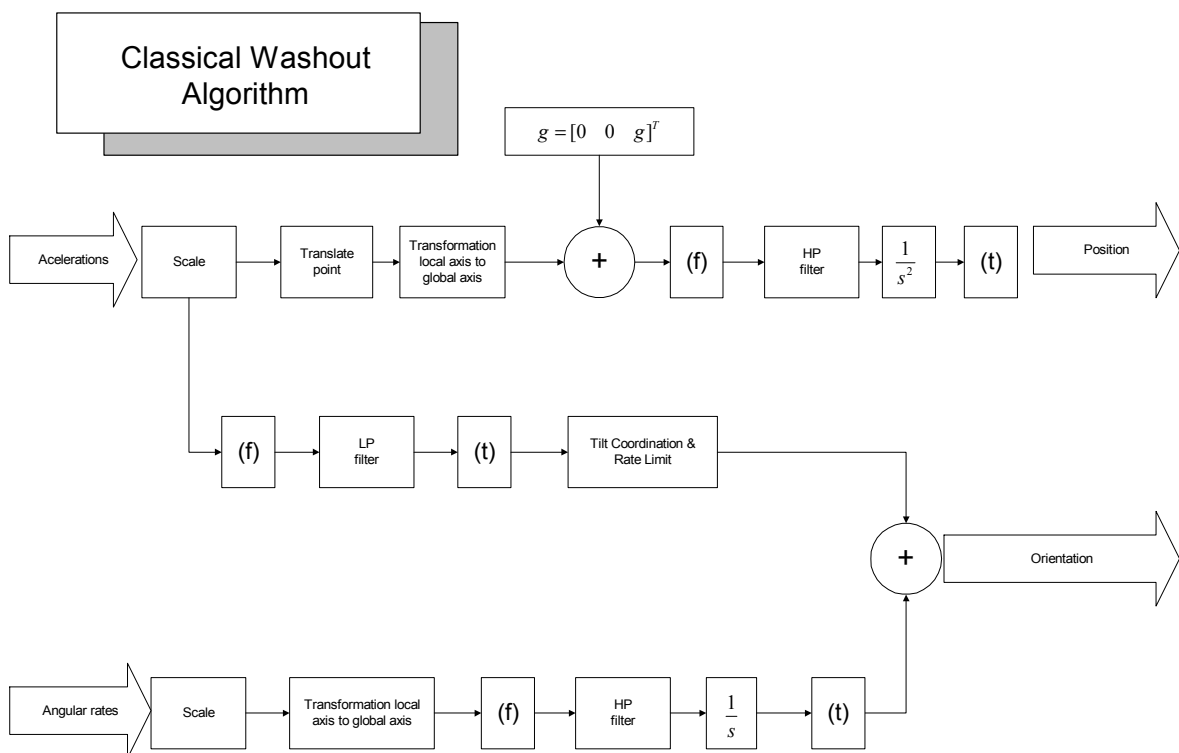


Figura 3-1: un modelo de Algoritmo de Washout

La teoría de washout clásico distingue entre tres etapas dentro del algoritmo. Tanto la primera como la segunda trabajan con las aceleraciones lineales como *input* mientras que la tercera etapa hace lo propio con los ángulos que recibe del simulador.

Observamos dos líneas de salida hacia la plataforma, ambas son necesarias para calcular el movimiento de la misma, son la posición y la orientación

La aceleración del vehículo no puede ser reproducida totalmente, un compromiso debe ser realizado entre la restitución de los índices inerciales y mantener la plataforma en su espacio de trabajo. La plataforma realmente vuelve a la posición central o de referencia en anticipación de nuestro próximo movimiento del simulador, sin que el conductor perciba esto.

Antes de implementar cada una de las partes se llevó a cabo un estudio y diseño de las mismas. El programa de control definitivo estará programado en 'c' pero antes de escribir el código que permitirá ejecutar las instrucciones se realizarán pruebas de las partes más críticas (filtros y filtrado de las señales de entrada) en *MatLab* para comprobar el correcto funcionamiento de las partes diseñadas.

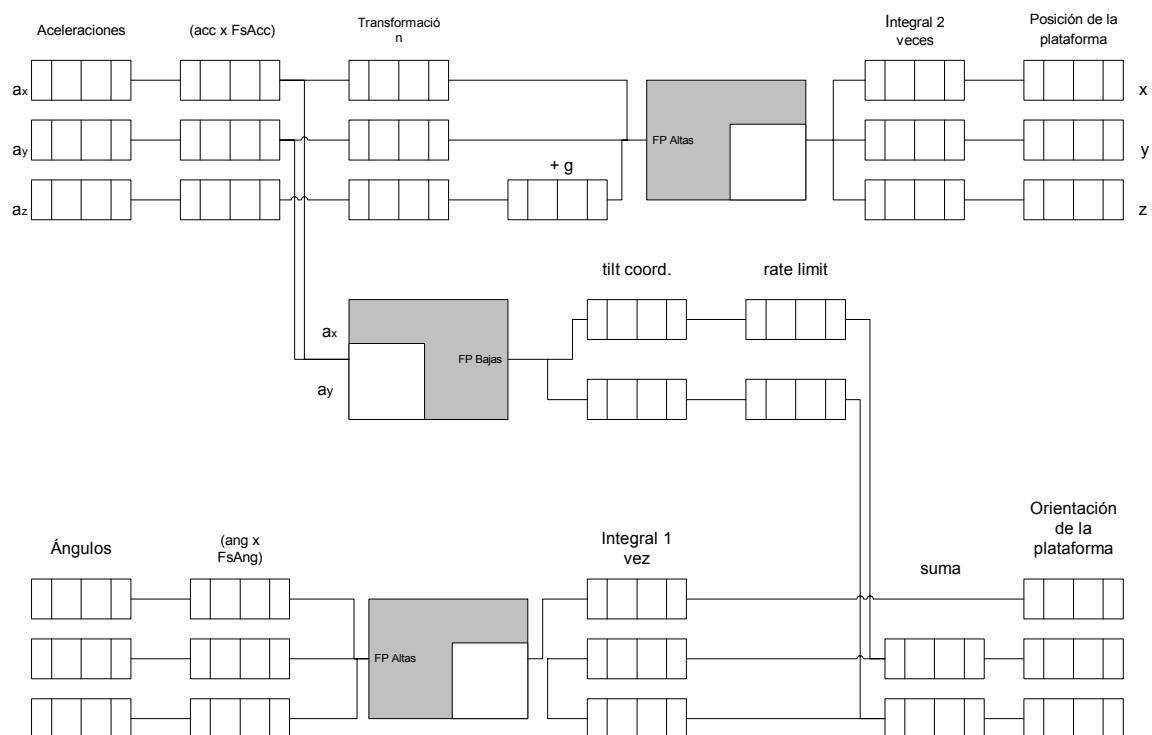


Figura 3-2: Señales y el algoritmo de Washout

La figura sobre estas líneas representa cómo una sección de la señal, muestreada para operar con fragmentos de un número determinado de datos, pasa por las distintas etapas que propone la teoría de *washout*.

¿Por qué es necesario el proceso de filtrado en el algoritmo? Las señales de entrada que recibimos son resultado de reacciones del conductor cada pocos milisegundos. Los movimientos que realiza la plataforma deben ser lo más coordinadamente posibles con el visual. El filtrado paso alto nos permite rebajar una señal que varíe lentamente de valor en un determinado número de muestras, es decir, que sea de baja frecuencia. Si hay pocas variaciones de valores registradas en esa ventana de muestras, la plataforma apenas debe moverse. Sin embargo, si se producen cambios la plataforma debe registrarlos.

En la etapa de *tilt-coordination* por el contrario nos interesa que la sensación de movimiento sea más "estable", menos explosiva y por tanto se propone un filtro paso bajo, interesan las bajas frecuencias.

Como se ha podido observar en el apartado 2.3.3, la información perteneciente a cada modelo o algoritmo es muy esquemática. Las publicaciones que hay al respecto hacen referencia a resultados a partir de ideas cuyos desarrollos no son puestos a disposición del consultante. En algunos casos ese contratiempo se podría salvar mediante suscripciones, pero la burocracia y la limitación temporal del proyecto hacia que la toma de decisiones sobre el desarrollo tuviese que avanzar sin este recurso.

A lo largo de los meses en los que este proyecto se desarrolló en Indra hubo diferentes propuestas a la hora de encarar la solución de este algoritmo. Fue en la etapa de análisis, estudio y documentación cuando se decidió ser autores del diseño de las "cajas" que componen el modelo. No se encontraron, a priori, información sobre los filtros u otros elementos que mejorasen el comportamiento de los canales. Por este motivo se planteó la opción de desarrollar el algoritmo como si de un sistema de tratamiento de señal discreta se tratase. Los resultados no convencieron a lo que se sumó una nueva fuente que permitió disponer de un esquema de filtros con los que trabajar, lo que llevó al planteamiento basado en la teoría de control. Esta opción es la que eligen por norma general los teóricos del sector aeronáutico.

3.2. Planteamiento basado en un sistema discreto

Cuando se comenzó a trabajar con este algoritmo ya se tenía sobre la mesa la base de datos con la que se debía trabajar (ver apartado 5.2) Esto condicionó cómo considerar los datos de entrada.

El algoritmo de *Washout* propone una serie de filtrados, escalados e integrales que pueden ser interpretados como módulos de un sistema de tratamiento de señal discreto. Nuestra señal resultado (posición y orientación) se obtiene al tratar una señal discreta (aceleraciones y ángulos)

Una de las limitaciones a las que nos enfrentamos es que estamos ante un sistema que tiene que funcionar en tiempo real. Reducir al mínimo la carga computacional es una prioridad para lograr reducir los tiempos de respuesta.

En un acercamiento inicial se proponía realizar a cada ventana de muestras su correspondiente transformación en frecuencia mediante una *fft* (*fast fourier transform*) para, a continuación, multiplicar por un filtro (en frecuencia) y posteriormente la respuesta volverla a transformar, pero esta vez al dominio temporal.

El hecho de realizar una transformada y una anti transformada implica una suma de operaciones derivadas de aplicar la *fft* que suponen un coste de tiempo elevado a la hora de obtener en milisegundos resultados. No se pedían resultados muy precisos, primaba más la velocidad con la que se servían datos hacia la plataforma de movimiento. La *fft* se dejaba de lado en este primer planteamiento por si se era capaz de lograr buenos resultados sin este recurso.

La alternativa a la multiplicación en frecuencia es la convolución en el tiempo. Aunque computacionalmente la multiplicación de la señal por el filtro es más eficiente no es así si tenemos en cuenta que con la convolución evitamos realizar las transformadas de Fourier que conllevaban más operaciones.

¿Qué elementos tenemos que diseñar para obtener nuestro sistema? Como podemos comprobar en el esquema del algoritmo necesitamos tres filtros (dos paso alto y uno paso bajo) así como un módulo de integración y otro que realice la convolución de dos señales.

Debido a la propiedad industrial por parte de Indra no se incluye en este documento el código desarrollado.

3.2.1 Diseño de un filtro digital

Se entiende por filtro aquel sistema lineal e invariante que permite el paso de las componentes de la señal existentes en un determinado intervalo frecuencial, y elimina las demás. De forma ideal, el módulo de la respuesta frecuencial del filtro toma un valor constante en el margen de frecuencias que queremos conservar, que se denomina banda de paso. El intervalo de frecuencias complementario al anterior en que la respuesta en magnitud es nula se denomina banda de rechazo o atenuada. [12-13]

Los cuatro filtros básicos, desde el punto de vista ideal del comportamiento del módulo de la respuesta frecuencial, según sea la posición relativa de bandas de paso y bandas atenuadas, reciben el nombre de paso bajo, paso alto, paso banda y elimina banda, dependiendo de la parte del espectro de frecuencias en la que se centra la banda de paso. En este proyecto se hace uso de los dos primeros.

El paso bajo se caracteriza porque deja pasar todas las componentes frecuenciales de la señal en el rango bajo de las frecuencias, por debajo de una determinada frecuencia de corte, siendo el resto de componentes atenuadas por el filtro. El filtro paso alto presenta el comportamiento complementario al paso bajo.

En el diseño de filtros digitales se pueden distinguir dos tipos básicos. Por un lado, aquellos en la que la respuesta al impulso del filtro tiene un número finito de muestras distintas de cero, lo que da lugar a la denominación abreviada de filtros FIR (*Finite Impulse Response*). Por otro lado, existen los llamados filtros recurrentes cuya respuesta al impulso tiene longitud infinita o filtros IIR (*Infinite Impulse Response*).

Aquí se requiere del uso de filtros FIR. Existen diversos métodos para el diseño de filtros FIR. Uno de ellos, el que se elegirá, es el de enventanado de la respuesta al impulso.

Parece más o menos intuitivo el pensar que si se toma la secuencia infinita de la respuesta $h(n)$ y se aísla una parte de ella, el resultado desde el punto de vista de la función de filtrado del sistema sería el mismo. Aunque a *grosso* modo eso es así, desde el punto de vista frecuencial se producen una serie de deformaciones en el espectro del filtro obtenido que nos llevarían a considerables errores a no ser por el uso de ventanas pensadas para este uso. Esas ventanas no son más que secuencias de longitud finita que tienen una respuesta frecuencial que permite que al ser multiplicadas por la función de transferencia utilizada el error no sea muy grande.

A partir de una función *sinc* en el dominio temporal se puede crear un filtro paso bajo ideal:

$$h_{pf}[n] = \sin\left(\frac{w_c \cdot (n - N/2)}{\pi \cdot (n - N/2)}\right)$$

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% filtro paso bajo ideal
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%hlp_i = wc/pi*sinc(wc/pi*(n-N/2));
for i=1:N
    if i==N/2
        hlp_i(i)= wc/pi;
    else
        hlp_i(i) = sin (wc*(i-N/2))/ (pi *(i - N/2));
    end
end
end

```

Figura 3-3: Código de un filtro paso bajo ideal diseñado

Aunque el programa que se ejecutará en la plataforma tiene que estar escrito en lenguaje *c*, se hizo uso de *Matlab* para probar las distintas soluciones propuestas para cada parte del algoritmo. Es el caso de los filtros que ocupa en esta parte de la memoria.

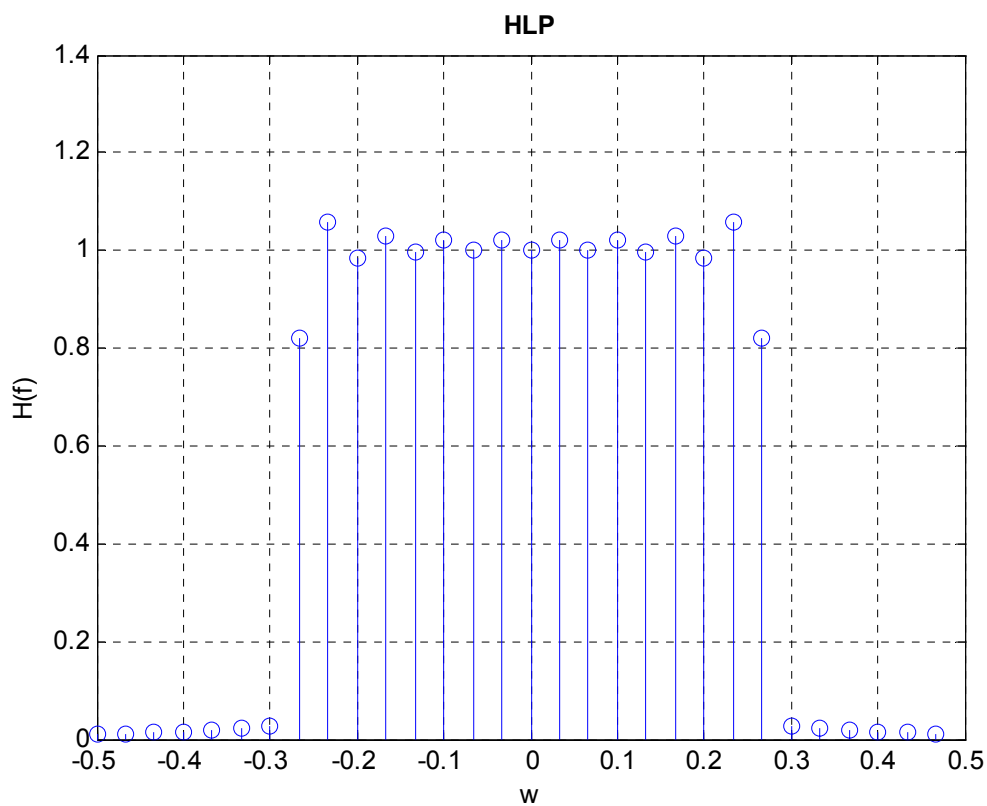


Figura 3-4: Filtro paso bajo dominio frecuencial ($F_s=1800$; $F_c=500$; $N=30$) diseñado

El enventanado consiste en el producto de la señal en cuestión por otra de duración finita. Esta señal de duración finita puede tener distintas configuraciones:

- ventana de Hamming
- ventana de Hanning
- un escalón
- etc.

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%   Aplicamos ventanas espectrales y normalizamos
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Ventana de Hamming
w = 0.54 - 0.46 * cos ((2*pi*n)/(N-1));

hlp = hlp.*w;
    
```

Figura 3-5: Código del enventanado por Hamming

Tras normalizar a uno, se obtiene el filtro paso alto operando $hhp = (-1).^n.*hlp$

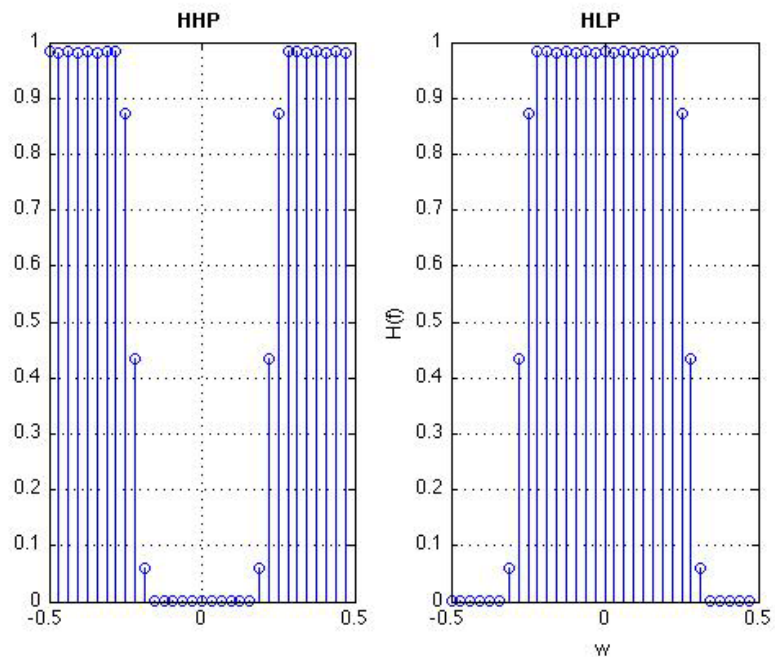


Figura 3-6: Filtro paso alto enventanado (N=30) diseñado

3.2.2. Diseño de un módulo de convolución

La convolución es una operación matemática que transforma dos funciones f y g en una tercera función que representa la magnitud en la que se superponen f y una versión trasladada e invertida de g . Se define como el sumatorio del producto de ambas funciones después desplazada una distancia n :

$$f[m] * g[m] = \sum_n f[n]g[m - n]$$

```
for k=1:nx
  for i=1:ny+1
    if (i-k>0 && i-k<(nh+1))
      y(i-1)=y(i-1) + x(k)*h(i-k)
    end
  end
end
```

Figura 3-7: Código diseñado en MATLAB para hacer una convolución

En *Matlab* existe una función que permite realizar convoluciones automáticamente si se le facilitan las dos señales implicadas en la operación. No es el caso de *c*. Había que desarrollar un fragmento de código que se obtuviera la misma salida. En primer lugar se hicieron pruebas en *Matlab* para finalmente a partir de código final realizar los cambios pertinentes en lo que a índices se refiere y mantener su funcionamiento como código en *c*.

3.2.3. Diseño de un módulo integrador

Antes de obtener la señal de salida con la posición de la plataforma, el valor de la señal corresponde a una aceleración, así que se tendrá que integrar dos veces para obtenerla en la dimensión que necesitamos.

Se optó en este planteamiento por usar las ecuaciones de la cinemática y a partir de ellas lograr la posición, ya que en cada instante de tiempo la aceleración es constante.

$$v - v_0 = \alpha \cdot (t - t_0)$$

$$x - x_0 = v_0 \cdot (t - t_0) + \frac{1}{2} \cdot \alpha \cdot (t - t_0)^2$$

$$v^2 = v_0^2 + 2\alpha(x - x_0)$$

3.2.4. Conclusión

Llegado a este punto se realizó una valoración del planteamiento y se optó por desarrollar otro diseño que se acercase más a la idea original de los creadores del algoritmo. Se realizaron pruebas en *Matlab* y no arrojaron unos resultados satisfactorios, no se cumplían las "bondades" de la teoría clásica de *Washout*, debido en parte porque hasta que no se hizo el siguiente desarrollo (apartado 3.3) no se desechó la idea de tratar sólo la muestra actual sin tener en cuenta sus predecesoras.

3.3. Planteamiento basado en teoría de control

Finalmente se planteó una alternativa que resultó ser a la postre la más adecuada para los objetivos. Una alternativa al diseño del sistema aplicando teoría de control al mismo, donde los módulos se encuentran desarrollados en el dominio de Laplace para facilitar las operaciones de filtrado e integración. Se partirá de los filtros de Grant & Reid [14].

3.3.1. Filtros propuestos por Grant & Reid

Se trata de filtros digitales con los siguientes parámetros. El más importante, ω , los valores de ω corresponden a las frecuencias de corte de los distintos filtros.

El coeficiente de amortiguación ζ es una medida que describe cómo después de una perturbación, las oscilaciones afectan a un sistema. El comportamiento de los sistemas oscilantes es de interés para la ingeniería de control que nos ocupa.

El valor del coeficiente de amortiguamiento ζ determina el comportamiento del sistema:

- Sobreamortiguado ($\zeta > 1$): El sistema vuelve (decae exponencialmente) al equilibrio sin oscilación.
- Amortiguamiento crítico ($\zeta = 1$): El sistema vuelve al equilibrio lo más rápido posible, sin oscilación. Aquí es hacia donde queremos que tienda el comportamiento.
- Subamortiguado ($\zeta < 1$): El sistema oscila (con una frecuencia ligeramente diferente que el caso no amortiguado), con la amplitud disminuyendo gradualmente a cero.

El sistema no sería estable si el amortiguamiento fuese menor que cero.

Para completar el diseño del sistema se decidió usar estos filtros. Habían sido probados experimentalmente por sus autores con éxito lo que se tuvo en consideración. Eso sí, el trabajo desarrollado en este proyecto decidió que se transformarían en el dominio de Fourier. Esta decisión implicaba que se iba a necesitar realizar *fft* y posteriormente *ifft*, así que habría que lograr un código que minimizase al máximo su carga computacional.

A continuación los tres filtros propuestos para completar el algoritmo. El primero de ellos un filtro paso alto, se aplicará a las aceleraciones lineales en el canal por el que posteriormente se obtiene la posición:

$$H_{Hip-f} = \frac{s^2}{s^2 + 2\xi_{hp} \omega_{hp} s + \omega_{hp}^2} \cdot \frac{s}{s + \omega_b}$$

$$|H_{Hip-f}(w)| = \frac{w^3}{\omega_b^2 \cdot (\omega_{hp}^2 - w^2)} \cdot \frac{1}{\sqrt{(1 + a^2)} \sqrt{(1 + b^2)}}$$

$$a = \frac{2w^2 \cdot \omega_{hp} \cdot \xi_{hp}}{\omega_{hp}^2 - w^2}$$

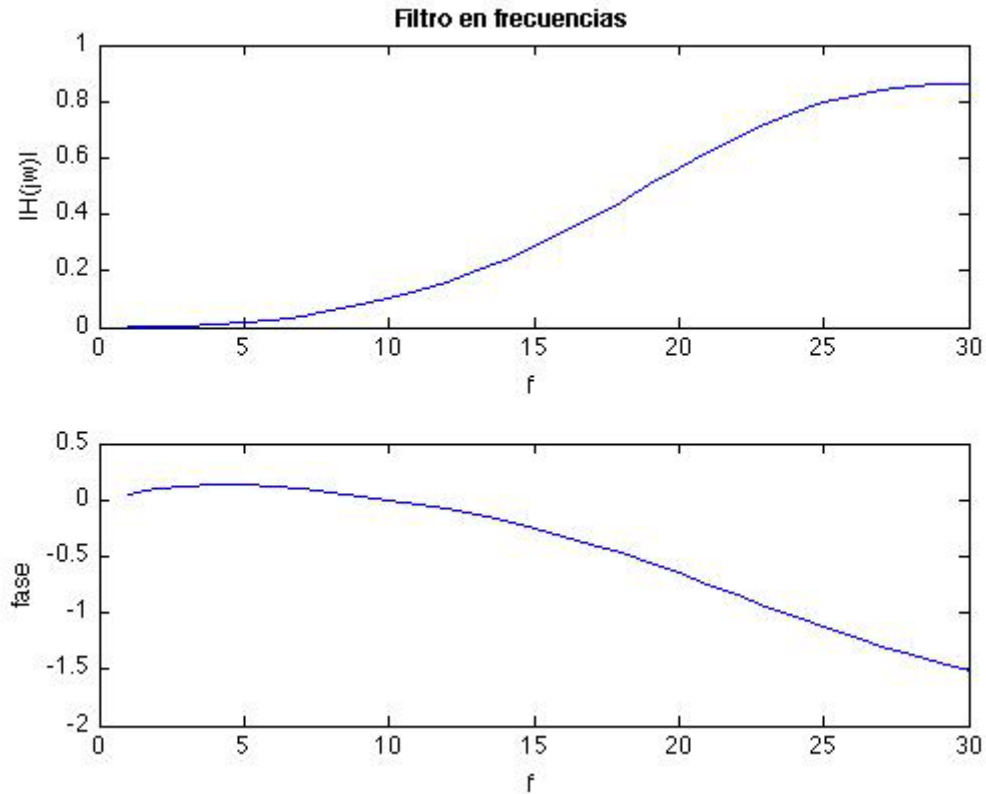


Figura 3-8: Filtro paso alto diseñado, aceleración

Sobre estas líneas una representación gráfica del filtro con los siguientes datos ($N=30$; $\zeta=0.09$; $\omega_h=900$; $\omega_b=200$)

Filtro paso bajo del *tilt-coordination* en el dominio de Laplace y en el de Fourier:

$$H_{Lp-f} = \frac{\omega_{lp}^2}{s^2 + 2\xi_{lp}\omega_{lp}s + \omega_{lp}^2}$$

$$|H_{Lp-f}(w)| = \frac{\omega_{lp}^2}{(\omega_{lp}^2 - w^2)} \cdot \frac{1}{\sqrt{(1+a^2)}}$$

$$a = \frac{2w \cdot \omega_{lp} \cdot \xi_{lp}}{\omega_{lp}^2 - w^2}$$

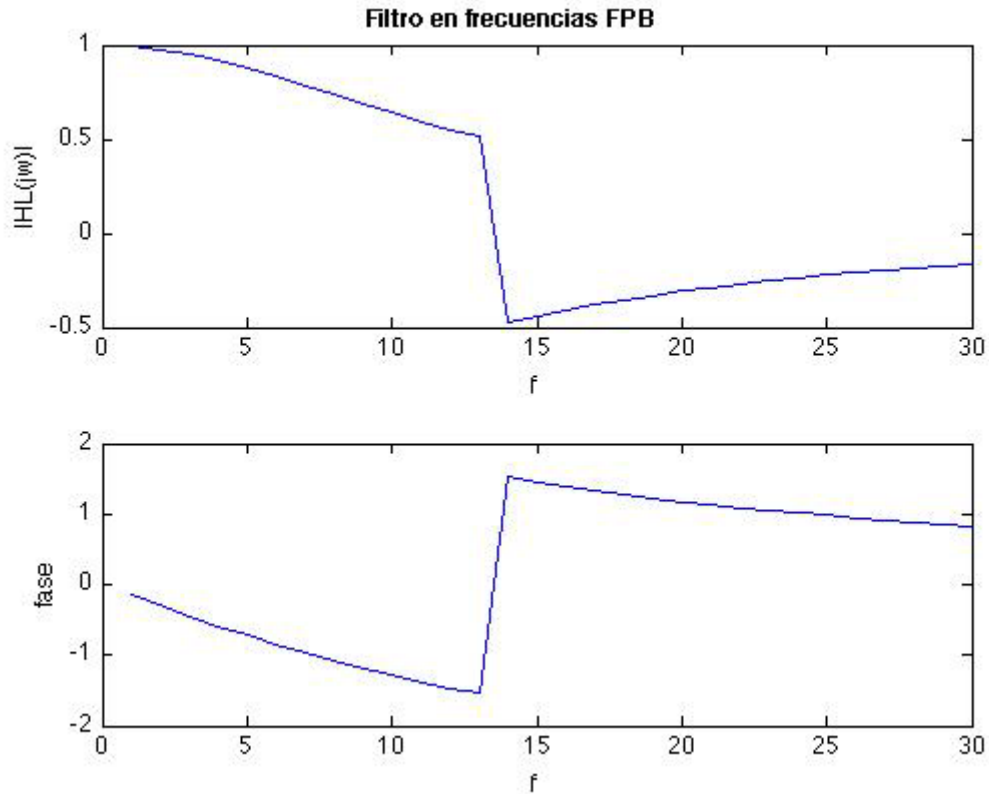


Figura 3-9: Filtro paso bajo diseñado

La gráfica anterior corresponde al filtro paso bajo con los siguientes datos ($N=30$; $\zeta=1$; $w_l=400$).

Llegados a este punto Grant-Reid proponían dos posibilidades de solución para el filtro paso alto que se aplican a las aceleraciones angulares, uno de primer orden y un segundo, por el que se apostó, de segundo orden (los autores de pruebas en este campo mostraban mejores resultados con filtros de este orden):

$$H_{Hp-\alpha} = \frac{s^2}{s^2 + \omega_{hp}s} \quad (\text{Filtro de primer orden})$$

El filtro elegido, de segundo orden:

$$H_{Hp-\alpha} = \frac{s^2}{s^2 + 2\xi_{hp}\omega_{hp}s + \omega_{hp}^2}$$

$$|H_{Hp-a}(w)| = \frac{w^2}{(\omega_{hp}^2 - w^2)} \cdot \frac{1}{\sqrt{(1+a^2)}}$$

$$a = \frac{2w \cdot \omega_{hp} \cdot \xi_{hp}}{\omega_{hp}^2 - w^2}$$

Como ejemplo la siguiente gráfica perteneciente al filtro anteriormente descrito, con los siguientes datos ($N=30$; $\zeta=0.5$; $wh=800$)

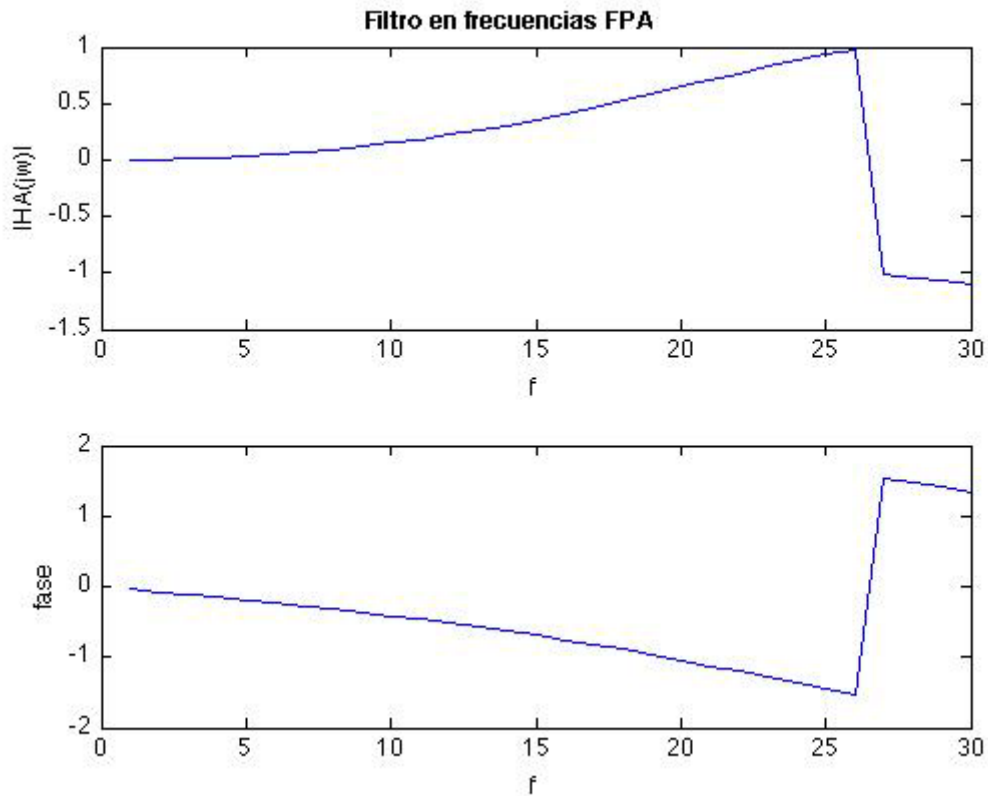


Figura 3-10: Filtro paso alto diseñado, ángulos

La elección de los parámetros de los filtros fue una tarea desarrollada una vez que el Sistema de pruebas (apartado 4) fue desarrollado. Se obtuvieron un rango de valores que hacían que los resultados que arrojaba el sistema de control fuesen aceptables. Estos valores al igual que el código desarrollado para implementar los sistemas es propiedad del cliente y no se hacen públicos en este documento.

3.3.2. Tratamiento de la señal de entrada y salida

La señal de entrada que nos proporciona el simulador es tratada por los distintos canales de nuestro algoritmo. Se planteó que la señal de entrada era causal, que no depende de sí misma en estados futuros.

En los canales del algoritmo la señal de entrada puede ser escala por un factor para mejorar los resultados finales.

Los filtros e integrales con las que se opera la señal están en el dominio de Fourier. Por tanto también la señal de entrada debe estar en ese mismo dominio. Dado que la entrada corresponde a una señal discreta no dependiente del tiempo sabemos que su correspondiente transformada de Fourier es la misma señal discreta. La transformada de Fourier de una constante es el resultado de multiplicarla por una delta *Dirac*.

3.3.3. FFT

La Fft (*Fast Fourier Transform*) es un algoritmo eficiente que permite calcular la transformada de Fourier discreta (DFT) y su inversa (*ifft*), a partir de una señal discreta $x[n]$ con N muestras (número potencia de dos) como entrada.

$$X(k) = \sum_{j=1}^N x(j)\omega_N^{(j-1)(k-1)}$$

$$x(j) = (1/N) \sum_{k=1}^N X(k)\omega_N^{-(j-1)(k-1)}$$

$$\omega_N = e^{(2\pi i)/N}$$

Se buscó un código *c* que permitió realizar *fft* y *iffts* que gestionara su propia memoria dinámica y cuyas entradas y salidas fuesen números complejos. Hubo que añadir módulos para obtener formas binomiales y de cálculo de módulo y fase para poder operar con ellas. Se realizaron pruebas comparando resultados con los que arrojaba la misma entrada en *Matlab*, logrando verificar el correcto funcionamiento de la transformada y de la antitransformada.

3.3.4. Diseño de un módulo integrador

Trabajar en el dominio de Laplace facilita la implementación del doble integrador:

$$I(s) = \frac{1}{s^2}$$

En el diseño del código, este multiplicador se tiene en cuenta en la tarea de filtrado, simplificando la función del filtro. Se observó que los filtros perdían sus propiedades al aplicar este factor así que se buscó una solución para la doble integración necesaria.

La solución que se propuso para este proyecto fue la regla del trapecio. La integración numérica se basa en la interpretación de la integral como el área encerrada bajo la curva. El método de integración numérica basado en Newton-Cotes consiste en el ajuste de un polinomio a un conjunto de puntos y luego, integrarlos.

$$\int_a^b f(x)dx \sim (b - a) \frac{f(a) + f(b)}{2}$$

Utilizaremos este método en dos ocasiones sucesivas para lograr transformar los datos de aceleración en coordenadas de posición.

Algunos autores proponen módulos de integración a la salida del canal que proporciona los datos de orientación a la plataforma. Este trabajo no valoró esta posibilidad porque la señales de entrada con las que se trabaja no lo requería.

3.4. Cálculo de la posición (traslación)

El primer paso en el proceso de toma de medidas es el de localizar los orígenes de los sistemas de coordenadas fijos de los cuerpos. El uso de un sistema de coordenadas centrales en la derivación de las ecuaciones del movimiento requiere que el origen de cada sistema de coordenadas debe medirse con el coche localizado en el centro de gravedad de los respectivos cuerpos.

Las aceleraciones lineales que se reciben del simulador están referenciadas respecto al centro de la estructura inferior. Recordemos que nuestro objetivo es el de conseguir la sensación más cercana a la realidad en el punto donde se encuentre el conductor. Es por tanto necesario lograr que la posición y la orientación de la plataforma de movimiento estén referidas a ese punto.

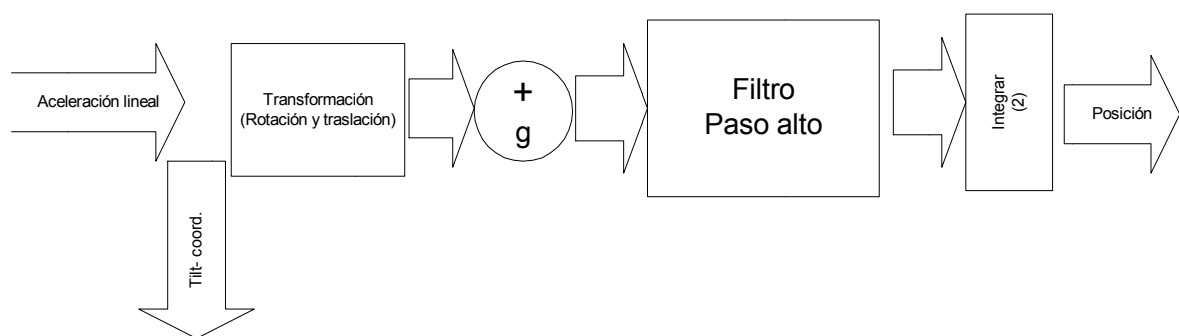


Figura 3-11: Diagrama de bloques del cálculo de la posición de la plataforma

Si se tiene un sistema de coordenadas móvil respecto de uno fijo, en tres dimensiones, y se desea dar la posición del sistema móvil en un momento dado, hay varias posibilidades de hacerlo. Una de ellas son los ángulos de navegación usados para describir la orientación de un objeto en tres

dimensiones. Se prefiere esta formulación a otras más elegantes por su baja complejidad computacional.

Los ángulos de navegación, llamados en matemáticas ángulos de Tait-Bryan, son tres coordenadas angulares que definen un triedro rotado desde otro que se considera el sistema de referencia. Se definen matemáticamente de forma similar a los ángulos de Euler, pero en vez de usar como línea de nodos el corte entre dos planos homólogos (por ejemplo el XY es el homólogo del xy), se utilizan dos planos no homólogos (por ejemplo XY e yz).

Muchas veces en ingeniería aeronáutica se utiliza el nombre de ángulos de Euler para hablar de estos ángulos. Se prefieren porque le asignan un ángulo de inclinación cero a un vehículo en horizontal, a diferencia de los ángulos de Euler, que le asignarían $\pi/2$. Estos ángulos también definen una rotación de forma única alrededor de cada uno de los ejes intrínsecos del objeto. Sin embargo, como ambas son formas de expresar la orientación de un cuerpo, existe una relación entre ellos; pudiéndose expresar unos en función de otros mediante una matriz de transformación.

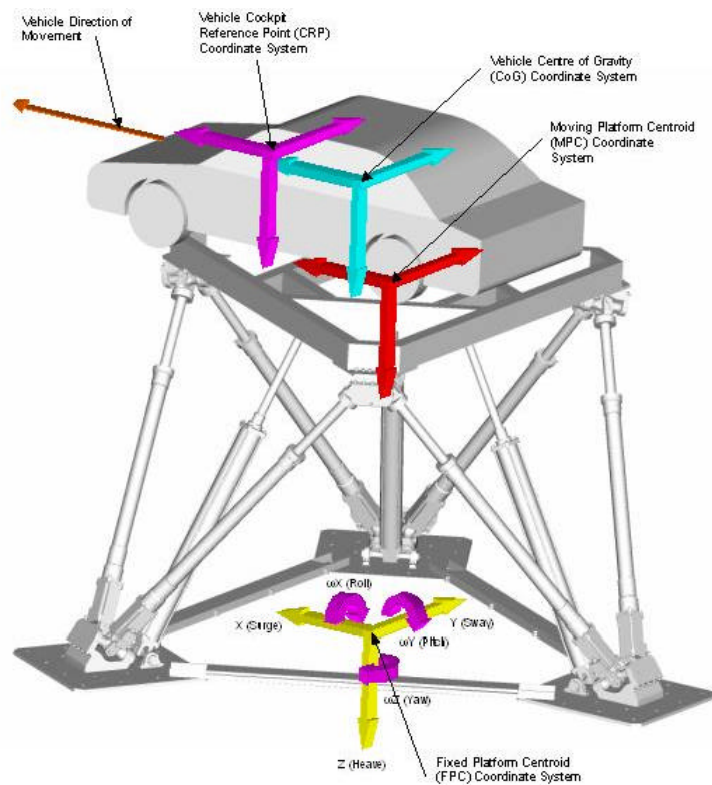


Figura 3-12: Gráfico coordenadas simulador terrestre

Dado un sistema de tres ejes fijos en el vehículo, llamados eje de inclinación (*yaw* en inglés), de cabeceo (*pitch*) y de alabeo (*roll*), existen tres rotaciones principales, normalmente llamadas igual que el eje sobre el que se producen, que permiten alcanzar el sistema del coche desde el sistema de referencia. Tienen que venir dadas y ser operadas en ese orden, ya que el resultado final depende del orden en que se apliquen.

- Cabeceo es una inclinación del morro del coche.
- Alabeo es la rotación intrínseca alrededor del eje frontal-trasera del coche.
- Guiñada es la rotación respecto de un eje vertical.

Son tres rotaciones intrínsecas, es decir, relativas al sistema móvil. Esto es útil por ejemplo cuando el conductor de un vehículo quiere describir una maniobra.

$$R(x, \text{heave/yaw}) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\psi & -\sin\psi \\ 0 & \sin\psi & \cos\psi \end{pmatrix} \quad \text{Matriz rotación eje x}$$

$$R(y, \text{pitch}) = \begin{pmatrix} \cos\vartheta & 0 & \sin\vartheta \\ 0 & 1 & 0 \\ -\sin\vartheta & 0 & \cos\vartheta \end{pmatrix} \quad \text{Matriz rotación eje y}$$

$$R(z, \text{roll}) = \begin{pmatrix} \cos\phi & -\sin\phi & 0 \\ \sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \text{Matriz rotación eje z}$$

Como se ha explicado antes, la multiplicación de las matrices tiene que hacerse en el orden contrario al que se efectúa la rotación:

$$R = R(z, \text{roll}) R(y, \text{pitch}) R(x, \text{yaw}) =$$

$$= \begin{pmatrix} \cos\phi \cos\vartheta & \cos\phi \sin\vartheta \sin\psi - \sin\phi \cos\psi & \cos\phi \sin\vartheta \cos\psi + \sin\phi \sin\psi \\ \sin\phi \cos\vartheta & \sin\phi \sin\vartheta \sin\psi - \cos\phi \cos\psi & \sin\phi \sin\vartheta \cos\psi - \sin\phi \sin\psi \\ -\sin\vartheta & \cos\vartheta \sin\psi & \cos\vartheta \cos\psi \end{pmatrix}$$

En líneas previas se planteaba que se quería trasladar al conductor los mejores resultados posibles. Para ello se tiene que saber que las sensaciones físicas que se sienten en un vehículo son diferentes dependiendo dónde nos situemos. Es por esta razón por la que se desarrolló en este trabajo, a parte de la rotación explicada antes, una traslación de los datos.

Se trasladan las aceleraciones desde el centro de masas de la plataforma (estructura superior) hasta el punto de referencia que será aproximadamente la posición donde se sitúa la cabeza del conductor.

Ese parámetro, el vector de posición, será diferente para cada vehículo porque la posición del puesto de conducción difiere entre unos y otros. Para este proyecto se calculó la distancia desde el centro de la plataforma hasta el reposacabezas del habitáculo del simulador de Indra.

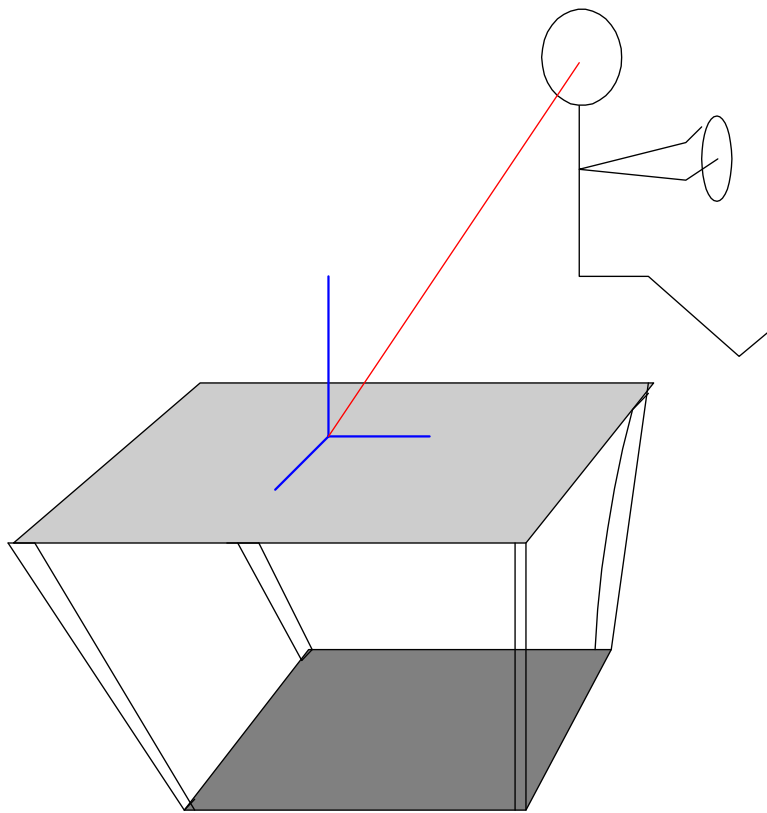


Figura 3-13: Punto de referencia

La ecuación para obtener el resultado que se requiere proviene de la teoría del sólido rígido para calcular la aceleración de dos puntos fijos.

$$\vec{a}^P = \vec{a}^0 + \vec{\omega} \times (\vec{\omega} \times \vec{P}) + \vec{\alpha} \times \vec{P}$$

Siendo:

\vec{a}^p La aceleración lineal en el punto de referencia escogido

\vec{a}^0 La aceleración lineal en el centro de masas

$\vec{\omega}$ La velocidad angular de la plataforma

\vec{P} El vector posición del punto de referencia escogido

$\vec{\alpha}$ La aceleración angular de la plataforma

Los ángulos, aceleraciones angulares y velocidades angulares necesarias para aplicar estas ecuaciones se obtienen a la vez que los parámetros de entrada (aceleraciones lineales).

Con la rotación y traslación de los datos referidos a la aceleración lineal referenciados correctamente se puede pasar a aplicar la rutina que propone el algoritmo para lograr el cálculo de la posición.

Tras aplicar a la señal un factor de escala se procede a sumar la constante de gravedad a la componente z de nuestra aceleración lineal.

Es el turno de aplicar el filtrado a la señal y posterior aplicación del módulo de integración.

3.5. Simulación de la gravedad (*tilt-coordination*)

El segundo camino o canal del algoritmo es conocido como *tilt-coordination*, permite, manteniendo las aceleraciones longitudinales y laterales, mediante la inclinación de la cabina ganar un vector de componente gravitatorio minimizando la sensación de aceleración continua. Con esto se obtiene unas fuerzas resultantes donde se alinea la gravedad con el vector fuerza específica total del vehículo simulado mediante la rotación de la base de movimiento.

El uso de un *tilt-coordination* puede mejorar la fidelidad del movimiento. Diversos estudios han analizado que la amplitud de la inclinación corporal podría aumentar la percepción de la aceleración lineal visualizada.

Este método se aprovecha de una de las características de los otolitos (ver apartado Sistema vestibular), no son capaces de distinguir la aceleración de traslación de la fuerza de la gravedad.

Inclinar la plataforma permite desarrollar nuevas componentes entre el eje X y/o el eje Y (depondrá también de la rotación). Estas componentes que se generan son percibidas por el

conductor como si de una aceleración de traslación se tratase y sirven para sustituir las que el conductor debería sentir en un vehículo real.

Esta inclinación puede ser interpretada por el sistema vestibular del conductor como una aceleración positiva o negativa cualquiera, dependiendo de la dirección de la inclinación. Eso si, el ratio de inclinación debe estar dentro del umbral de los canales semicirculares.

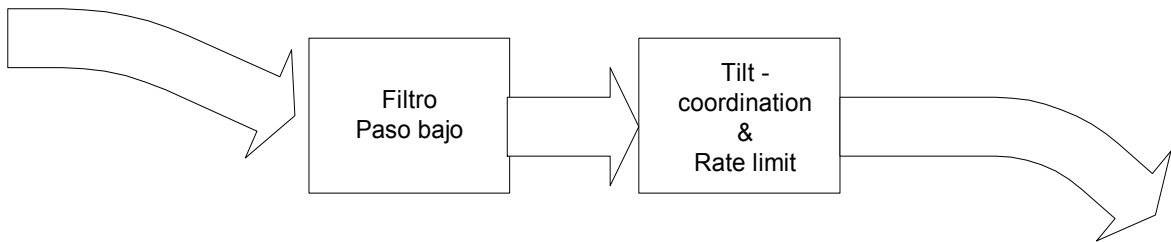


Figura 3-14: Diagrama *Tilt-coordination*

Las componentes *tilt-coordination* resultantes dependen de los vectores (módulo y dirección) de las aceleraciones longitudinales a_x y a_y . La aportación de estas ayuda a que las sensaciones que se tienen que transmitir al conductor sean lo más cercanas a la realidad.

Para llevar a cabo la misión de este canal se debe operar con componentes de baja frecuencia de las aceleraciones. Se van a "crear" unas falsas entradas, unas componentes de fuerza de gravedad que en la plataforma son logradas por la inclinación de la cabina de simulación, de tal modo que las componentes x e y de las g's corresponden con a_x y a_y . Pero antes el filtro paso alto bloquea las bajas frecuencias de las señales de movimiento, las cuales tienden a producir un gran desplazamiento hacia delante.

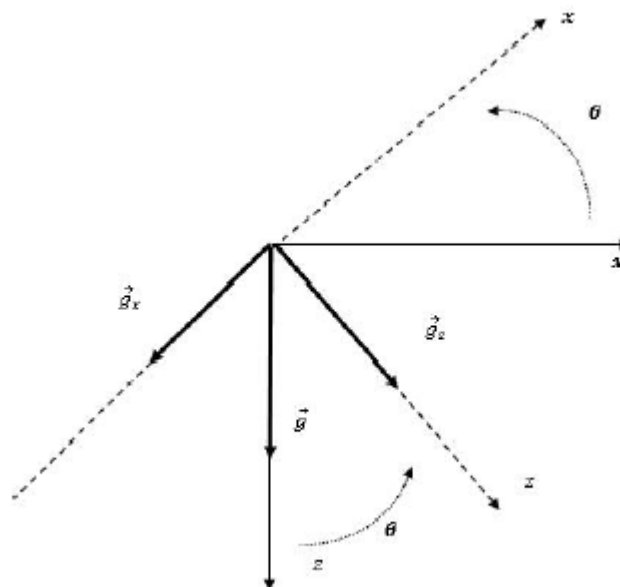


Figura 3-15: Fuerzas simuladas en el eje X.

$$a_x = g \cdot \sin \vartheta \Leftrightarrow \sin \vartheta = \frac{a_x}{g} \Leftrightarrow \vartheta = \sin^{-1} \left(\frac{a_x}{g} \right)$$

$$\boxed{\vartheta = \sin^{-1} \left(\frac{a_x}{g} \right)}$$

$$a_y = g \cdot \cos \vartheta \cdot \sin \phi \Leftrightarrow \sin \phi = \frac{a_y}{g \cdot \cos \vartheta} \Leftrightarrow \phi = \sin^{-1} \left(\frac{a_y}{g \cdot \cos \vartheta} \right)$$

$$\boxed{\phi = \sin^{-1} \left(\frac{a_y}{g \cdot \cos \vartheta} \right)}$$

Cada plataforma tiene unas limitaciones físicas que hay que tener en cuenta en este punto del algoritmo. Los resultados no pueden sobrepasarlas. De ser así el conductor podría experimentar falsas sensaciones debidas a estas componentes. Los valores de las aceleraciones para calcular el pitch y el roll deben estar entre unos límites. Ver apéndice A.1.

La limitación de este índice es aplicado para prevenir la sensación del índice de rotación angular asociado con la inclinación.

3.6. Cálculo de la orientación (rotación)

El planteamiento original del algoritmo de *washout* contaba en esta etapa con las aceleraciones angulares como señales de entrada pero en el caso tratado el algoritmo se beneficia de un simulador cuyo sistema de proceso proporciona directamente los ángulos (*heave/yaw, pitch, roll*).

Esto permite realizar el filtrado del *input* sin necesidad de llevar a cabo ningún tipo de transformación como sucedía con las aceleraciones lineales ni incluir un posterior módulo de integración ya que la salida son ángulos igualmente, estos determinan la orientación de la plataforma.

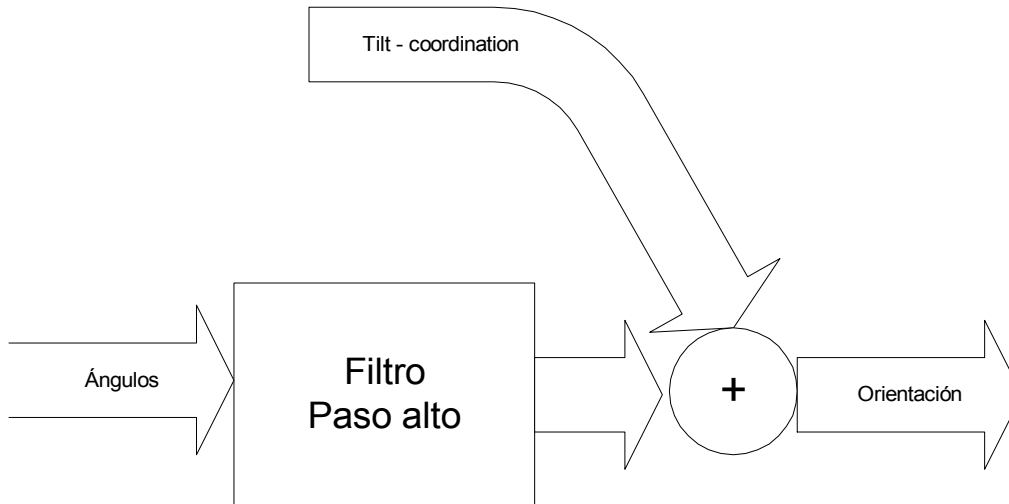


Figura 3-16: Diagrama de bloques del cálculo orientación de la plataforma

Como se puede observar en la imagen de los módulos de este canal, se aplica a la entrada el filtro paso alto específico diseñado en el apartado 3.3.1

Una vez que se ha filtrado la señal, proceso ya explicado en el apartado 3.2, se sumará a la señal analizada en este punto el resultado de la etapa de *tilt-coordination*. Realizado este paso se tendrá unos ángulos que serán enviados a la plataforma y que determinan su orientación.

4. DESARROLLO DEL SISTEMA DE PRUEBAS

4.1. Características

La segunda parte de este proyecto consiste en diseñar y desarrollar un sistema que permita probar el algoritmo implementado. La interfaz gráfica programada para este proyecto, como requisito fundamental, debe permitir observar la representación gráfica de las señales de entrada y sus correspondientes resultados en la salida.

Los resultados mostrados, así como la longitud de las señales, dependerán de la configuración de los parámetros del algoritmo como hemos visto en apartados anteriores. Por esa razón será necesario dar la posibilidad al usuario de cambiar estas configuraciones. Estas nuevas combinaciones permitirán al cliente ver cómo afectan estas variaciones al comportamiento del sistema de control de la plataforma.

El sistema de pruebas desarrollado en este proyecto será usado por profesionales del departamento de simulación, no estando dirigido para el cliente final del simulador.

Una vez finalizado este proyecto se entregó al cliente la información necesaria para el buen uso del software desarrollado. [15-16]

4.2. Diseño e implementación

En el apartado tecnologías a utilizar se hacía referencia al uso de *Qt* para llevar a cabo esta tarea. Sobre *c++*, *Qt* es una herramienta que permite el diseño de una interfaz gráfica atractiva para el usuario. El programa que se generará en este proyecto contará con barra de menús, de estado y de herramientas donde se podrán activar las opciones de configuración exigidas. La apariencia es similar a la que tienen los programas ejecutados bajo *Windows*.

4.2.1. Primera versión

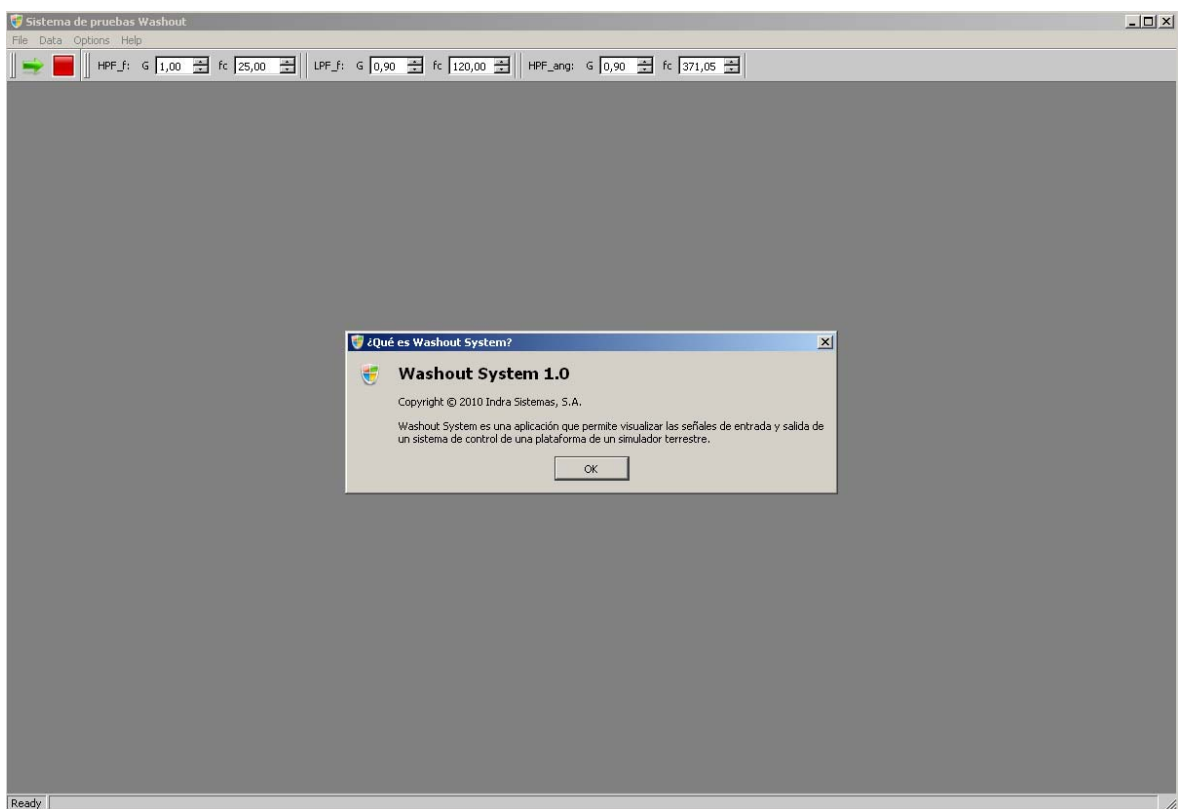


Figura 4-1: Interfaz Sistema de pruebas Washout

Configuración de parámetros generales:

- Estos son frecuencia de muestreo, muestras en la entrada, muestras en la salida, gravedad, vector posición del conductor respecto al centro de la plataforma y finalmente los límites que tiene la plataforma respecto al pitch y al roll.
- Estos datos deberían ser fijados en un principio, antes de ejecutar el algoritmo.

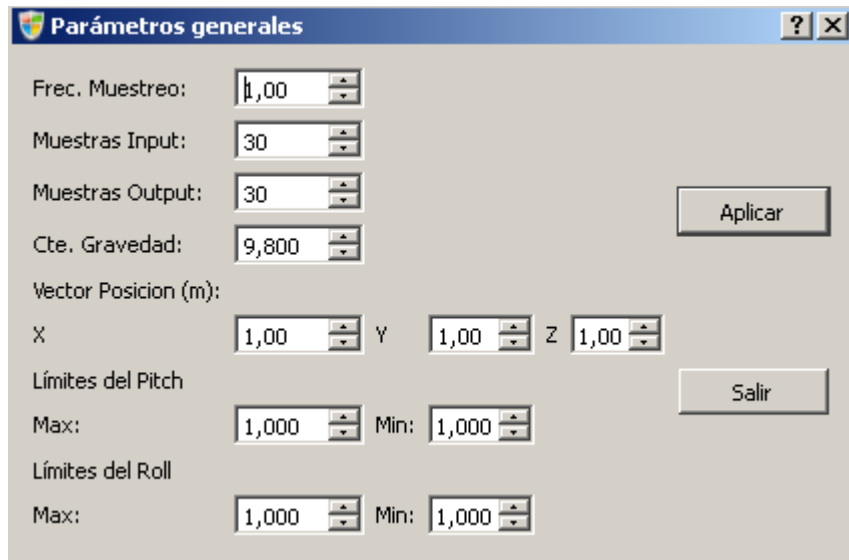


Figura 4-2: Ventana de ajustes parámetros generales

Configuración de parámetros a ajustar:

- En la barra de herramientas señalada encontramos la opción de configurar los filtros del algoritmo. Podemos ajustar tanto la ganancia como la frecuencia de corte de los tres filtros del algoritmo.

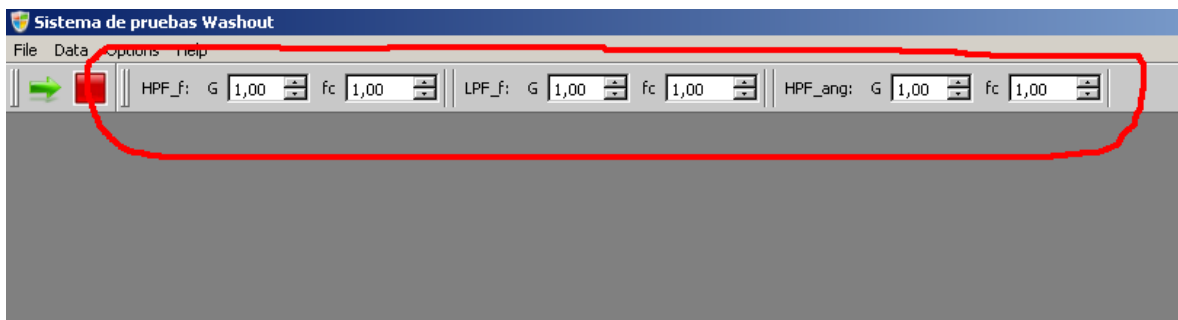


Figura 4-3: Tool bar de ajuste de frecuencias de corte y ganancias

Selección de ventanas a mostrar:

- Para el estudio del comportamiento de las entradas y las salidas, este *dock* de configuración permite seleccionar las ventanas de las señales que queramos estudiar.

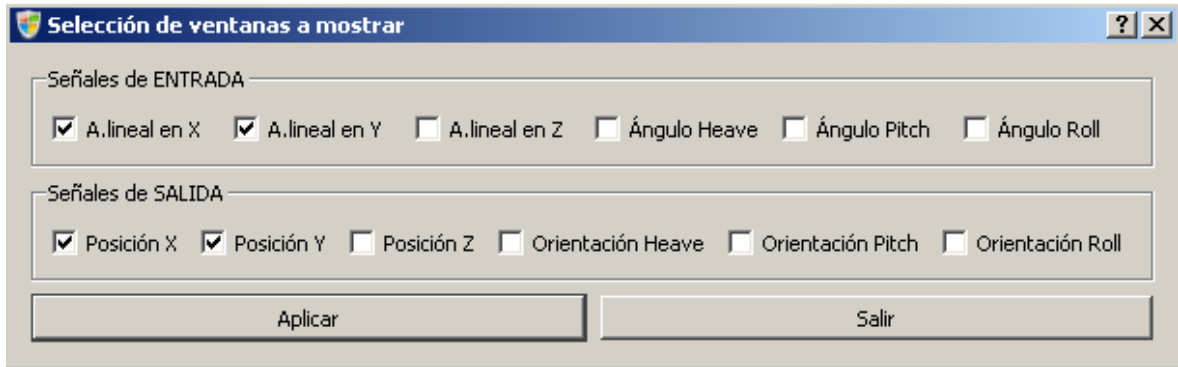


Figura 4-4: Selección de ventanas a mostrar

Ejecución del algoritmo:

- una vez que se tienen todos los parámetros completados se puede ejecutar el algoritmo pulsando el botón de la flecha verde que está en la barra de herramientas recuadrada en la imagen. En caso de que se desee reiniciar los valores, el botón rojo que aparece en la imagen hace efectiva esa función. Tras esto, si se vuelve a ejecutar el algoritmo se obtendrá las entradas de nuevo del comienzo de la base de datos.

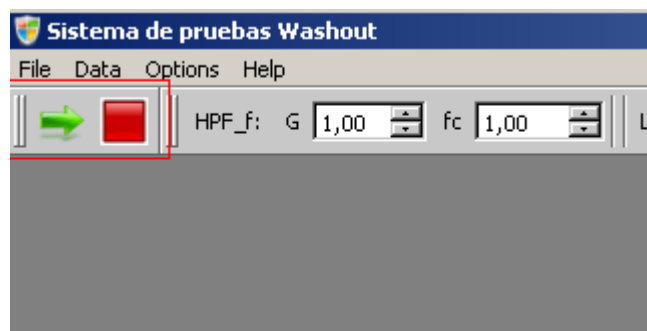


Figura 4-5: Botones de inicio de ejecución y reseteo

Selección de ficheros:

- Dentro del menú *Data* se encuentra la posibilidad de seleccionar el nombre de los ficheros donde se recogen los datos de entrada y salida:

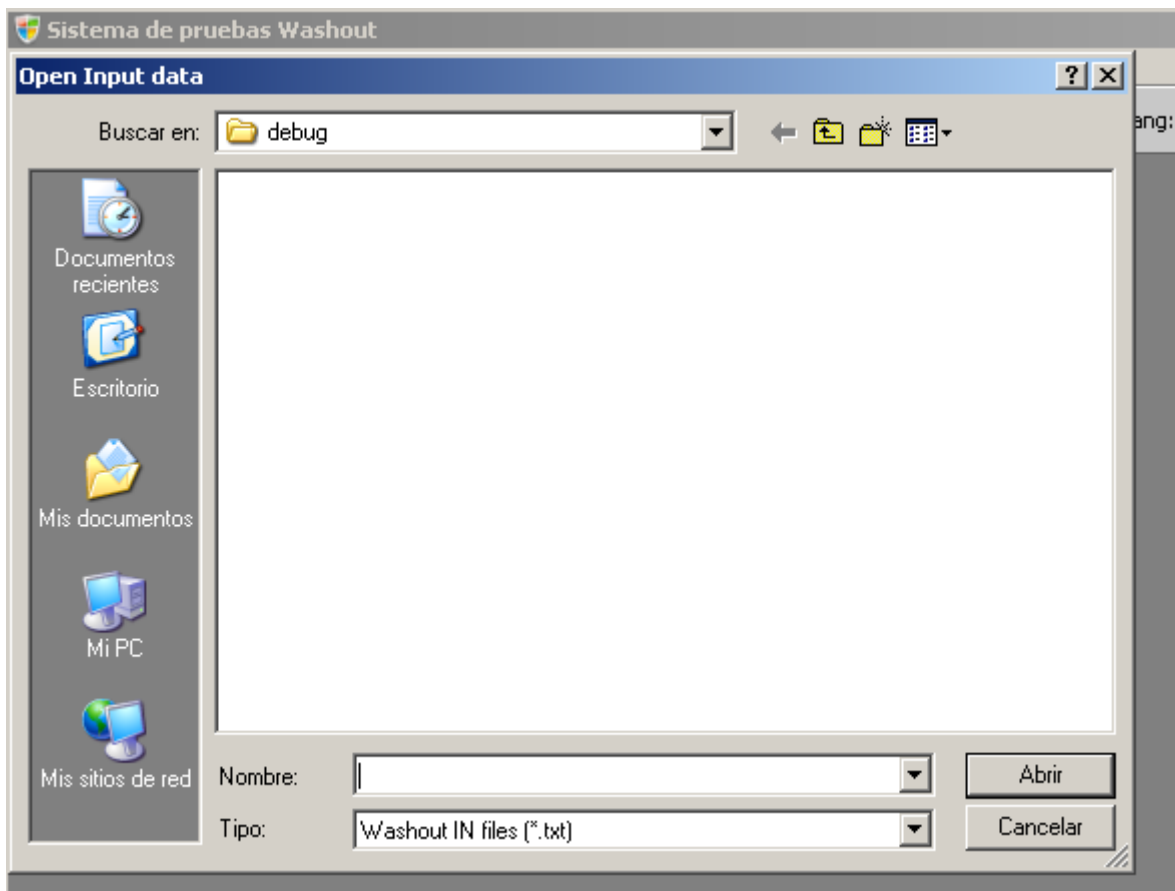


Figura 4-6: Ventana de selección de la base de datos

Cosas a tener en cuenta:

- Con Qt no hay que liberar memoria, si la jerarquía está clara, el mismo compilador se encarga. Si el padre termina su función los hijos son liberados automáticamente.

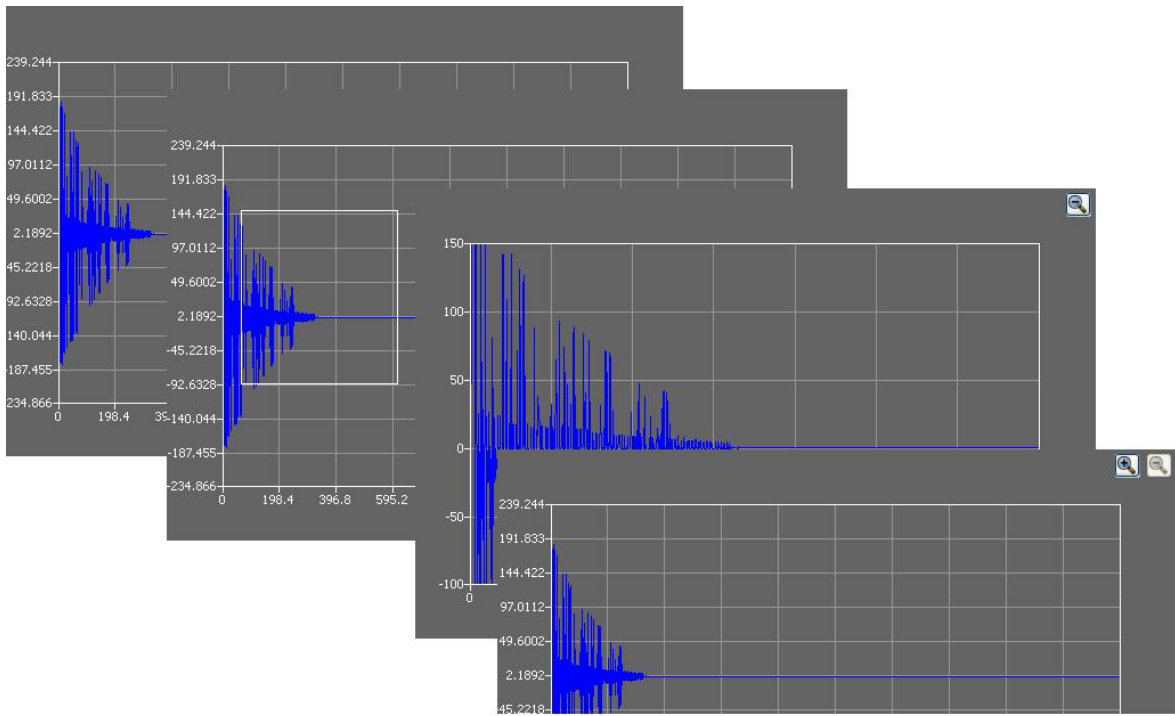


Figura 4-7: Uso del zoom

Dada una gráfica se puede ampliar la zona que se elija de la misma. A continuación aparecerá un icono que permitirá restaurar la gráfica. Si se pulsa, volviendo a la imagen original, surgirá un nuevo icono que permitirá volver a la imagen ampliada anteriormente.

4.2.2. Mejoras en el sistema de pruebas

Hasta aquí la primera versión del software creado. Una vez que las pruebas con el código de control empezaron a realizarse surgieron nuevas necesidades. Había que mejorar la capacidad de ajuste del sistema. Para ellos se incluyeron parámetros nuevos que, en un primer momento, se había valorado que fuesen constantes.

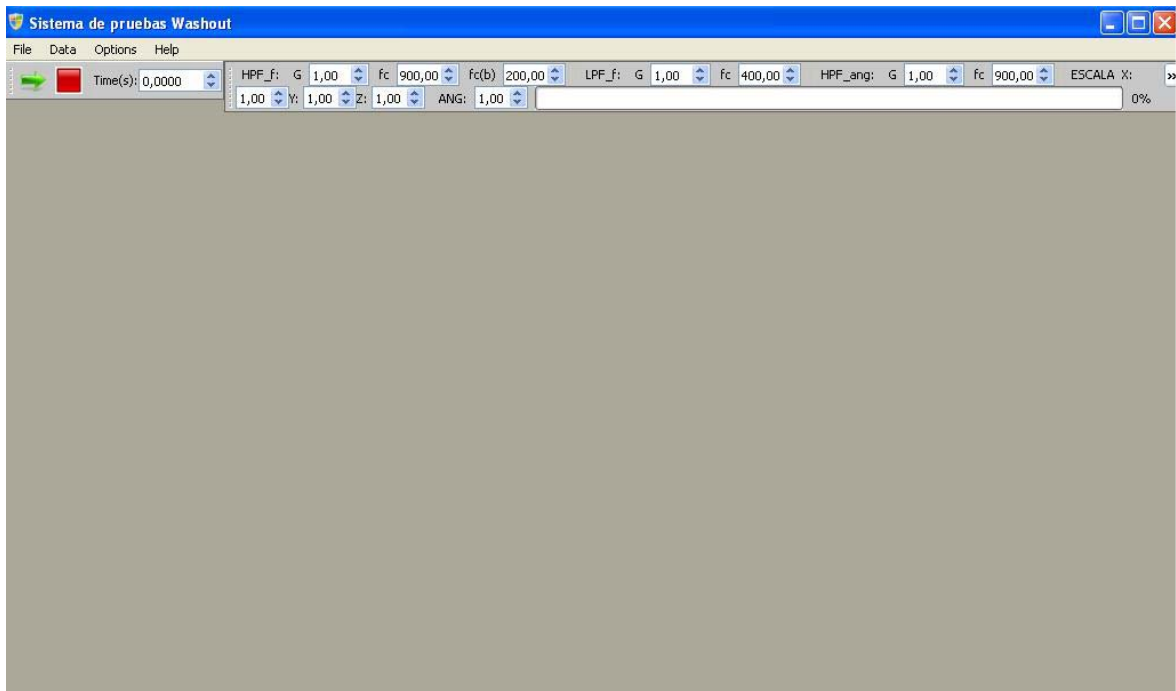


Figura 4-8: Nueva barra de ajustes

Ahora los filtros se pueden modelar con mayor precisión. Ya no sólo se cuenta con la posibilidad de variar la ganancia de los mismos si no que también se pueden escalar el valor de las entradas.

Para realizar las pruebas del sistema de control era importante contar con un elemento, *Time(s)* en la barra, que nos indicase el instante de tiempo desde que empezó la simulación en el que se encuentra el testeo. Gracias a este mecanismo también se puede acceder al instante de tiempo deseado.

En la imagen podemos encontrar una barra blanca que finaliza en un "0%". Se trata de una barra de progreso. Su implementación fue un valor añadido de especial utilidad cuando tras las primeras pruebas se cambió la gestión de los datos de entrada (ver apartado 5.2) Ahora el PC tardaba más tiempo en realizar los cálculos y la barra representa el progreso de los mismos. Así el usuario sabe que el sistema de pruebas está realizando su trabajo.

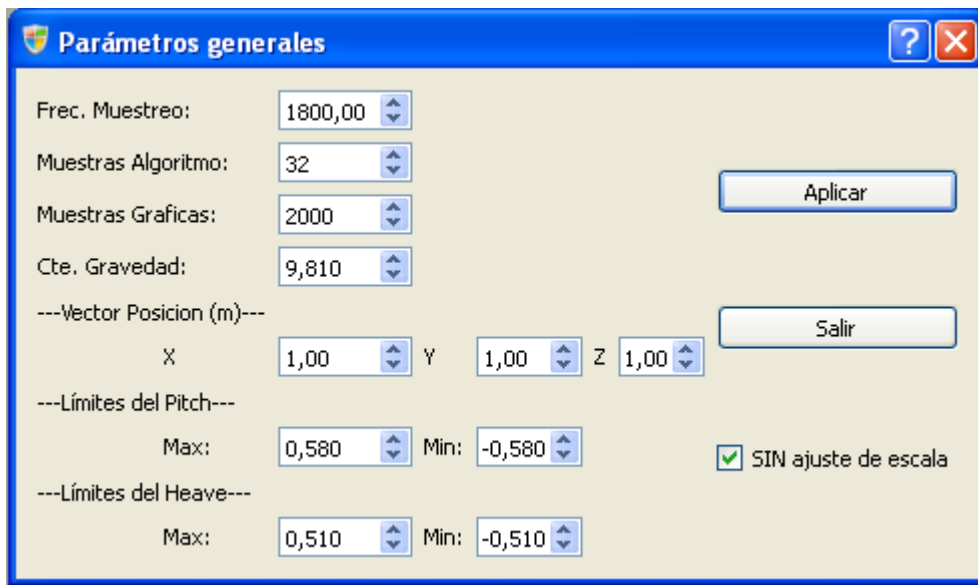


Figura 4-9: Nuevo ventana de parámetros generales

Cuando se comenzó a usar el software se decidió mejorar la ventana que rige los parámetros. Por un lado se pretendía hacer más clara. Lo referente a las muestras de entrada/salida tuvo que ser modificado. Cuando se imprime por pantalla la gráfica para realmente poder observar el comportamiento que tiene la señal hay que al menos mostrar un segundo. Esto se debe a la gran cantidad de muestras que genera el simulador.

$F_s = 1800 \text{ Hz} \rightarrow 1 \text{ muestra cada } 0.56 \text{ ms}$

Para obtener al menos 1 segundo de muestras, necesitamos 2000 muestras (1.11 s)

Respecto a las muestras que se toman a la entrada se fijó en 32 por el uso de la fft , el número de muestras debía ser múltiplo de 2 (2^n)

Este hecho tiene como consecuencia que realmente a la salida se representarán 1984 muestras (0.99 s) ya que 62 módulos de 32 muestras es el número más cercano a 2000. No habría muestras que dibujar si el programa no tuviese en cuenta este tipo de detalles.

Otra de las mejoras fue habilitar la posibilidad de que todas las gráficas relacionadas con una señal dejen de ser auto escalable. Es decir, que las gráficas mantengan siempre los mismos valores máximos y mínimos. Si por el contrario esta activa la opción, dependiendo de los máximos y mínimo de la señal (máximos y mínimos locales de ese tramo de representación) la gráfica se dibuja dentro de ese rango.

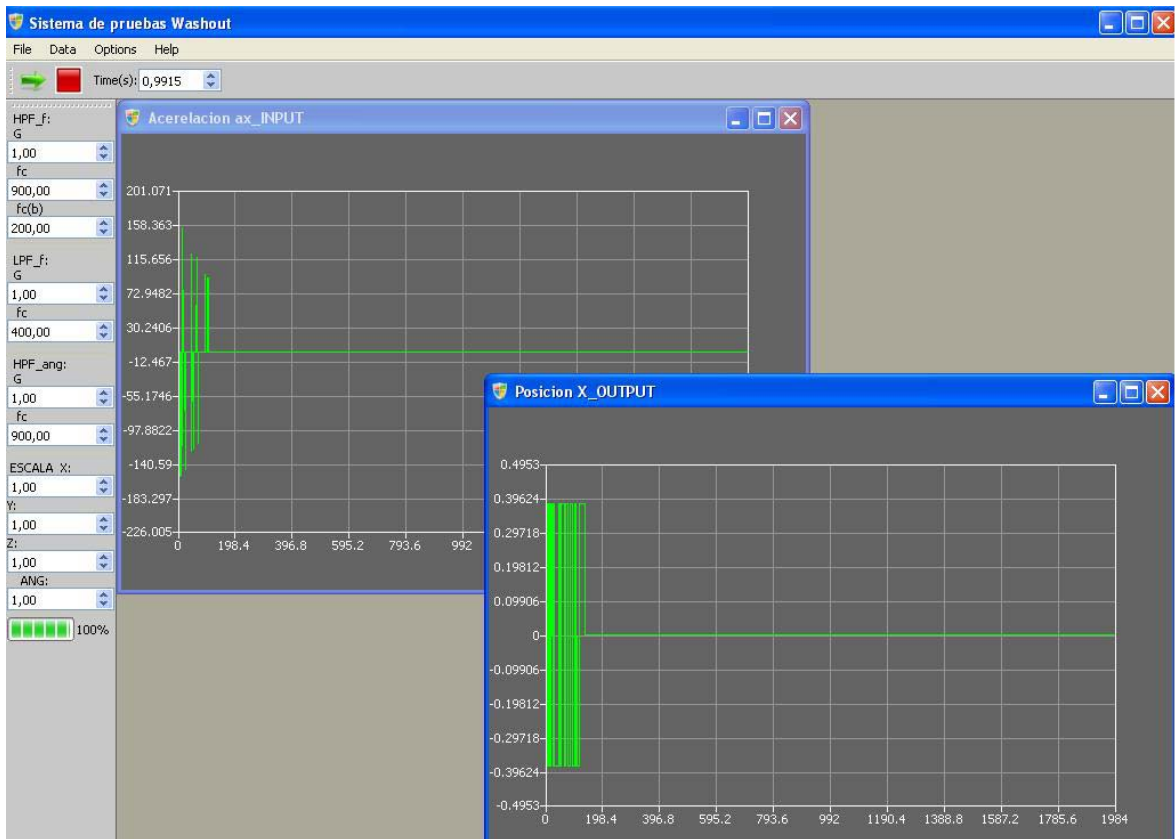


Figura 4-10: Ejemplo de funcionamiento

El ejemplo de funcionamiento de la figura 4-8, representa en las gráficas contenidas en él el primer segundo de funcionamiento del simulador.

Llegado a este punto el sistema de pruebas está listo para realizar su función.

5. INTEGRACIÓN Y PRUEBAS DE LOS SISTEMAS

5.1. Integración

Una vez que el algoritmo de control y el sistema de pruebas han sido terminados es el momento de fusionar sendos códigos. En esta etapa hay que plantear las dependencias que tiene uno con el otro para lograr una correcta integración.

El código desarrollado para la plataforma de movimiento tiene distintos parámetros que cuando se prueben en la plataforma ya serán prácticamente definitivos. Para cargarlos se hará uso de ficheros de texto con esta información. Sin embargo en esta fase de pruebas esos parámetros van a ser ajustados y debe facilitarse esa labor al usuario. Es por esto que esta información será tomada del interfaz de usuario creada y del resultado de la interacción con el mismo. Es decir, las características de los filtros, su ganancia y sus frecuencias de corte.

El algoritmo de control ha sido diseñado para funcionar de un modo continuo en la plataforma, como continuo será el flujo de información que recibirá. Pero ese modo de funcionamiento no nos interesa para realizar las pruebas ya que se quiere variar parámetros, probar con distintas configuraciones etc. Esto supondrá unos resultados distintos por cada modificación para una misma entrada. Por tanto la salida del algoritmo debe ser distinta ya que se pretende analizar esos cambios en las gráficas que el sistema de pruebas va a representar con estos datos.

5.2. Pruebas con la base de datos

5.2.1. Introducción

Algunas características de conducción deben ser tenidas en consideración a la hora de realizar ajustes en el sistema de control del movimiento. Habrá que saber que la teoría de control para plataformas de simuladores fue desarrollada inicialmente para aviones y que las variaciones de aceleración de un vehículo son algunas veces más agresivas que las observadas en una aeronave.

Conducir un vehículo tiene lugar dentro de un escenario de tráfico que puede crear en algunas ocasiones situaciones complejas. El conductor tiene que estar además más atento al control del coche que un piloto de aviones.

La base de datos fue elaborada a partir de los resultados del simulador de vehículo de policía que tiene Indra Sistemas en sus instalaciones. Estos resultados provienen de las maniobras que el conductor realizó. En concreto se trata de una aceleración con un posterior giro y frenazo.

0.000555556	0	11.6138	0	0	1	0	-0	0	0
1981.62	-2104.92	0.498376	-6.49726e-006	-3.32896e-008					
-11.6138	0	6.59872	-20904.7	0.00512343	-1.11889e-006				
-0.000315656	-0.00119313	-0.00179427	-1.13637	-2.14763					
7.25077	-2045.45	0	0	0	0	0			
0.00111111	0	11.5652	0	0	1	0	-0	0	0
1981.62	-2104.92	0.491937	0.0112024	0.0487776	-11.5651				
0	87.6133	87.6466	0.00512343	-2.89598e-006	-				
0.000934078	0.00120877	-0.00321463	-1.08996	4.31843	-2.98994				
83.5291	0	0.0046191	-14.6333	-2.23777e-009	-				
6.31312e-007									
0.00166667	0	11.4691	0	0	1	0	-0	0	0
1981.62	-2104.92	0.485539	-0.0973571	0.138571	-11.4679				
-173.85	162.363	175.251	0.00512618	-6.76702e-006	-				
0.00151612	0.00867961	0.00641235	-1.00545	13.475	16.5226				
152.201	0	0.0659484	-14.572	-8.02973e-009	-				
2.49947e-006									
0.00222222	0	11.3739	0	0	1	0	-0	0	0

Figura 5-1: Extracto de la base de datos

La base de datos, como se puede observar en la imagen, aporta gran cantidad de información del simulador. Datos como:

- Instante de la muestra
- Uso del claxon
- Velocidad mostrada
- Encendido del motor
- Se pisa el acelerador
- Se pisa el freno
- Se pisa el embrague
- Movimiento del volante
- RPM
- Marcha engranada
- Posición (recorrido)
- Velocidad lineal
- Aceleración lineal
- Ángulos
- Aceleración angular
- Velocidad angular
- Etc

El correcto tratamiento de estos datos por el sistema de control permite obtener los resultados que mostraremos a continuación.

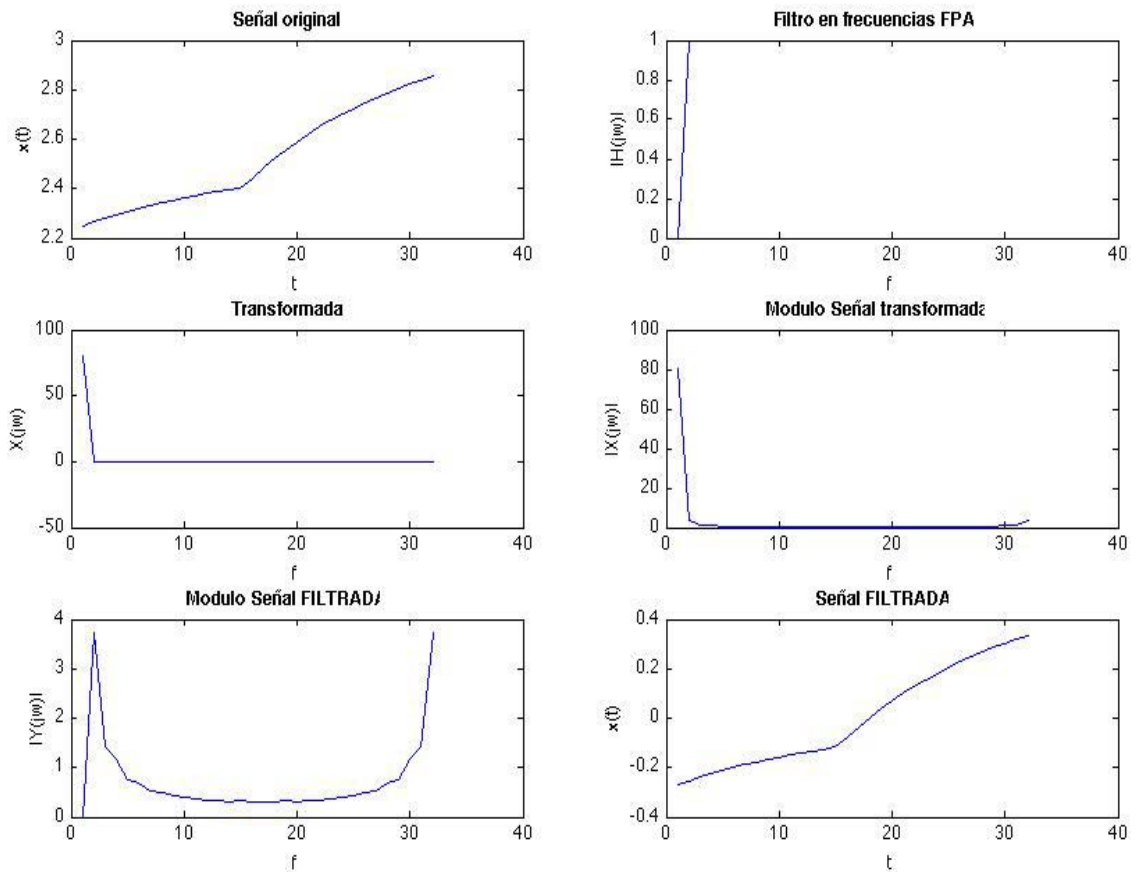


Figura 5-2: Primeros datos

Tras analizar los resultados de las gráficas sobre estas líneas, se observó que la salida no se correspondía con lo esperado. El comportamiento no era el previsto atendiendo a la teoría. La solución a este contratiempo estaba en el tratamiento de la señal de entrada.

En un primer planteamiento se trataban los datos avanzando de N muestras en N muestras. No se tenían en cuenta los valores anteriores, tan sólo el actual. Esto es un punto importante ya que la posición del conductor, las aceleraciones sufridas con anterioridad etc. si que influyen en el instante posterior. Dos situaciones a tener en cuenta:

- No es lo mismo acelerar y frenar que frenar y acelerar.
- Si tratamos de girar a un lado y posteriormente cambiamos el sentido.

Son situaciones que plasman la necesidad de tener en cuenta no sólo la muestra actual sino las anteriores también.

Por tanto, a la vista de los resultados se varió la forma de tratar los datos de entrada, los datos se presentarán en N muestras pero en cada instante se tienen en cuenta los N-1 valores anteriores. Ahora si, la señal en el estado actual depende de estados en el pasado.

Hubo que modificar consecuentemente el código. El que está integrado en el sistema de pruebas, hasta que haya suficientes muestras en los instantes iniciales, se programa para rellenar con ceros.

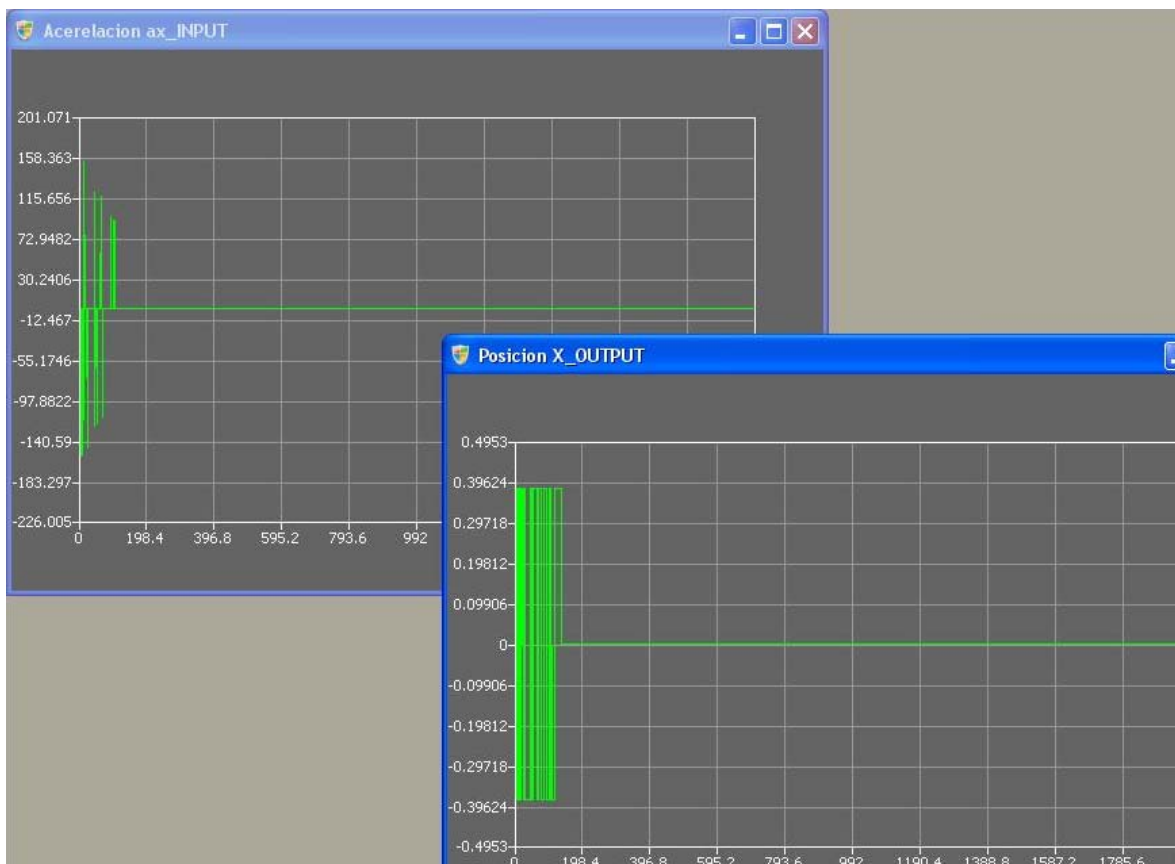


Figura 5-3: Ejemplo de funcionamiento

Se puede observar como inicialmente los valores fluctúan rápidamente debido a la puesta en funcionamiento y calibración de la plataforma.

Como se apuntó al inicio de este apartado la prueba de conducción consistió en una progresiva aceleración para finalmente pisar el freno. La duración de la misma son 30 segundos. Pero de ese medio minuto habrá que analizar el comportamiento del sistema de control justo donde se produce la frenada. Esto es así porque, si no se produce grandes variaciones la posición de la plataforma debe permanecer estable, neutra.

5.2.2. Pruebas del cálculo de la posición

Se escogieron 4 momentos para comprobar el correcto funcionamiento de la teoría de *Washout*:

- 1) A los 16 segundos se produce un cambio. Mayor presión en el pedal del acelerador provoca una aceleración mayor. Luego se mantendrá constante.

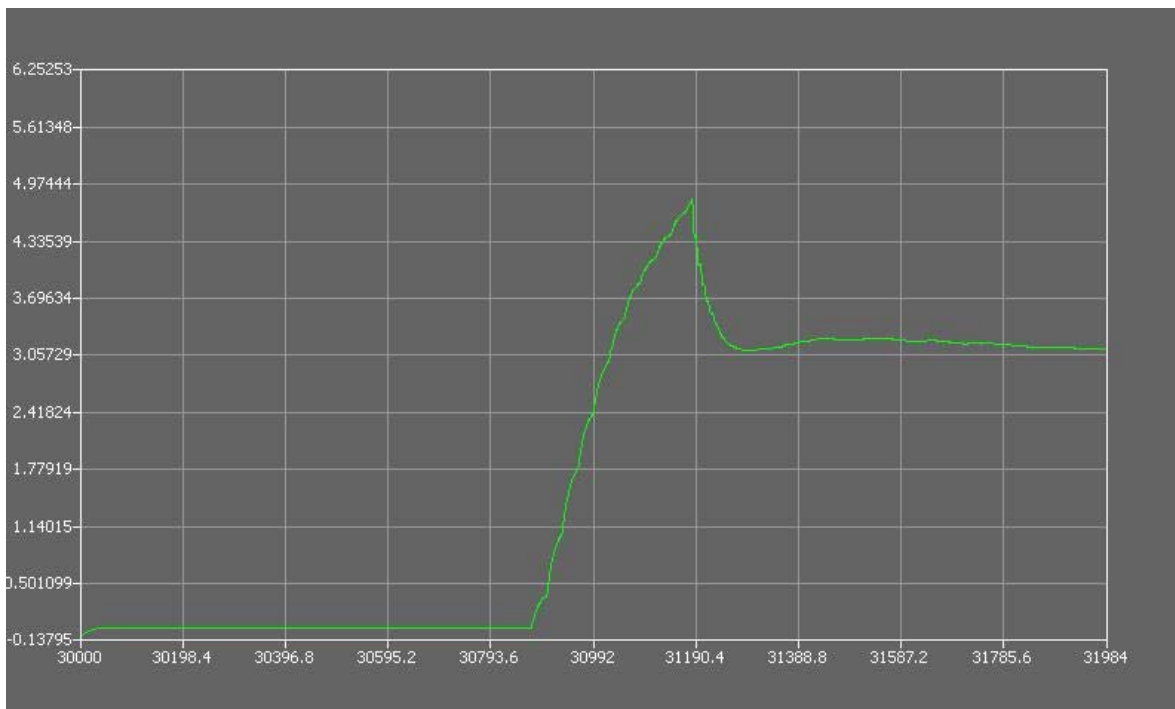


Figura 5-4: Grafica prueba ax (1)

Observamos la salida:

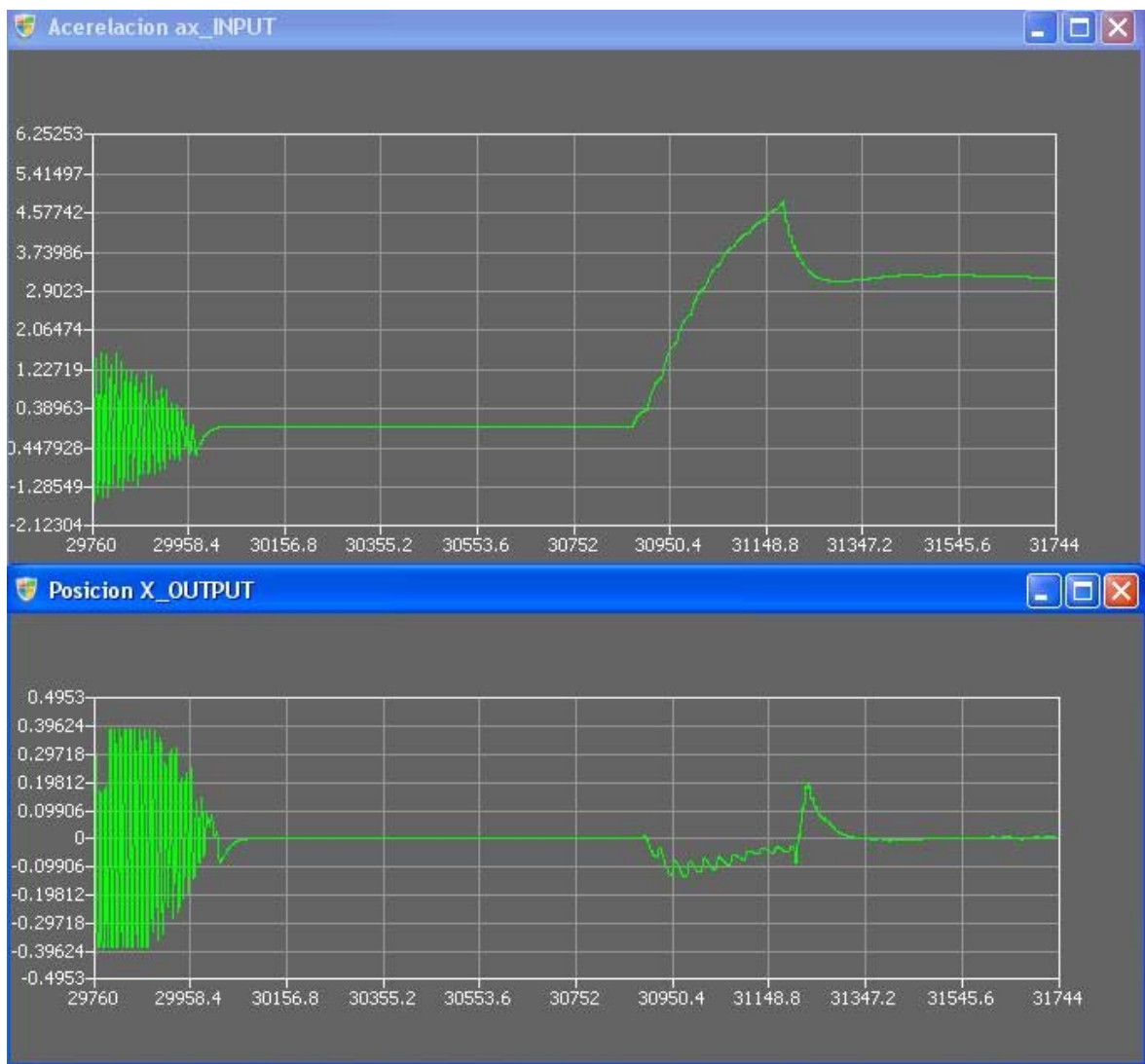


Figura 5-5: Grafica prueba ax y salida pos X (1)

- 2) Antes de los 19 segundos se ha producido una progresiva deceleración (se acelera menos) pero justo en este instante se produce una aceleración negativa.



Figura 5-6: Grafica prueba ax (2)

La salida:

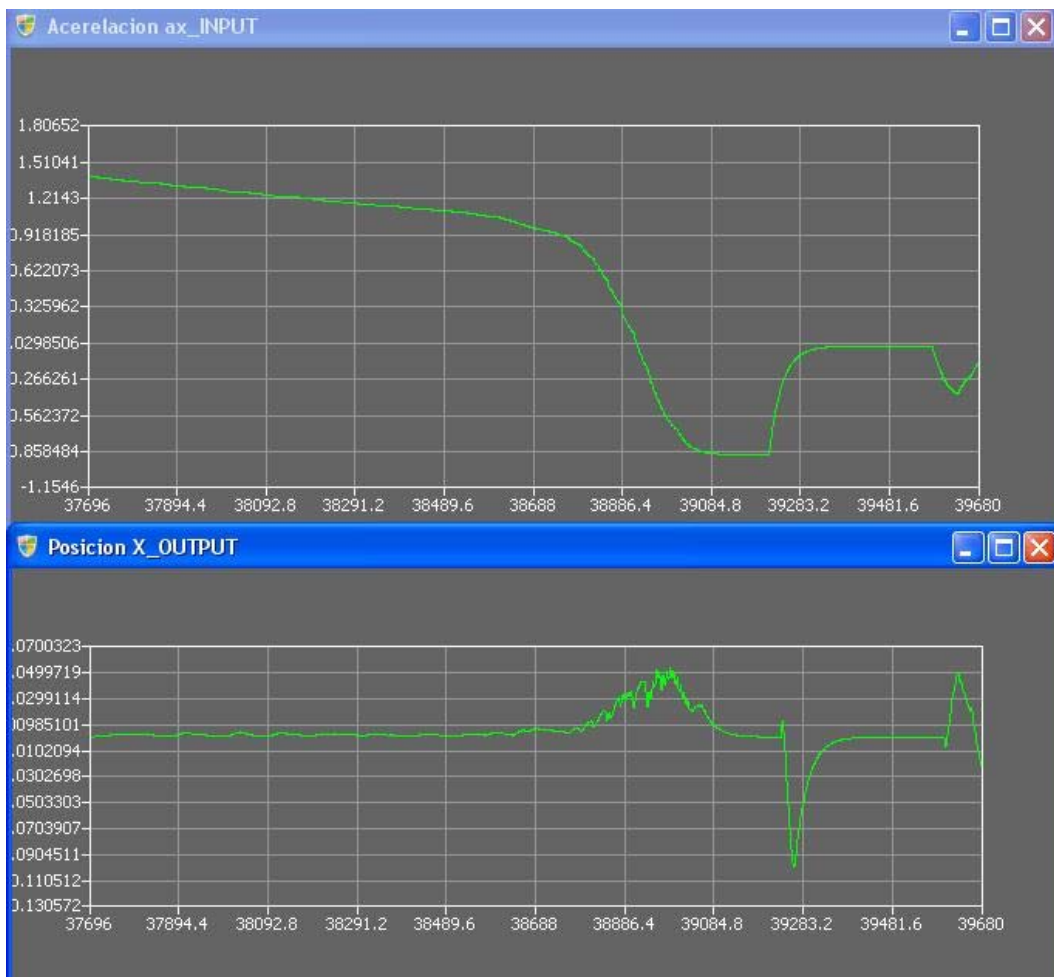


Figura 5-7: Grafica prueba ax y salida pos X (2)

3) Se observa una fuerte aceleración a continuación.

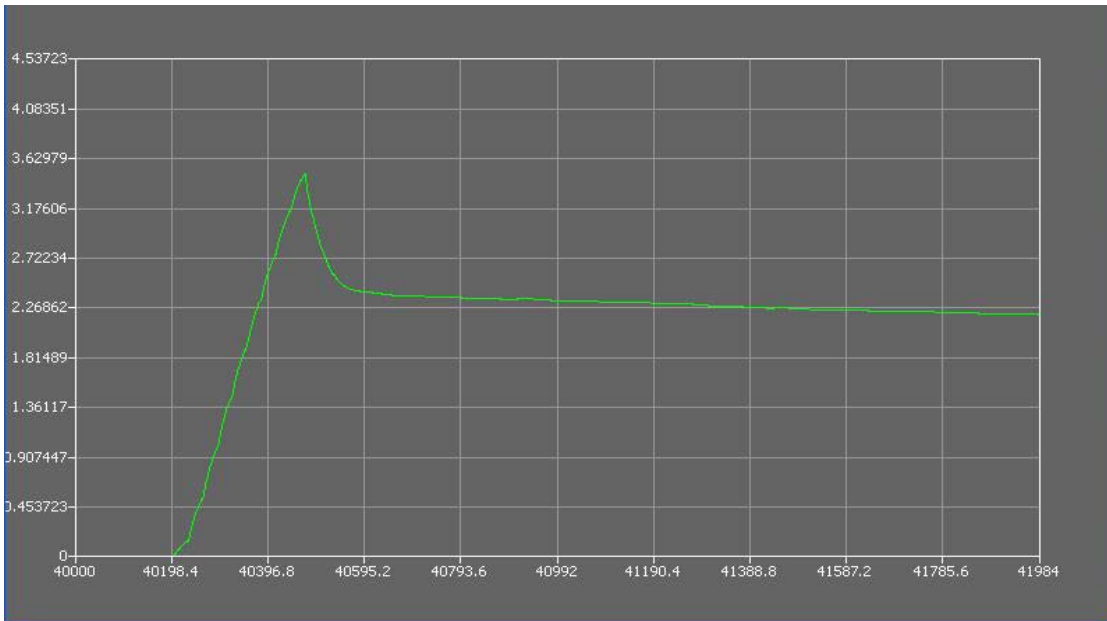


Figura 5-8: Grafica prueba ax (3)

La salida:

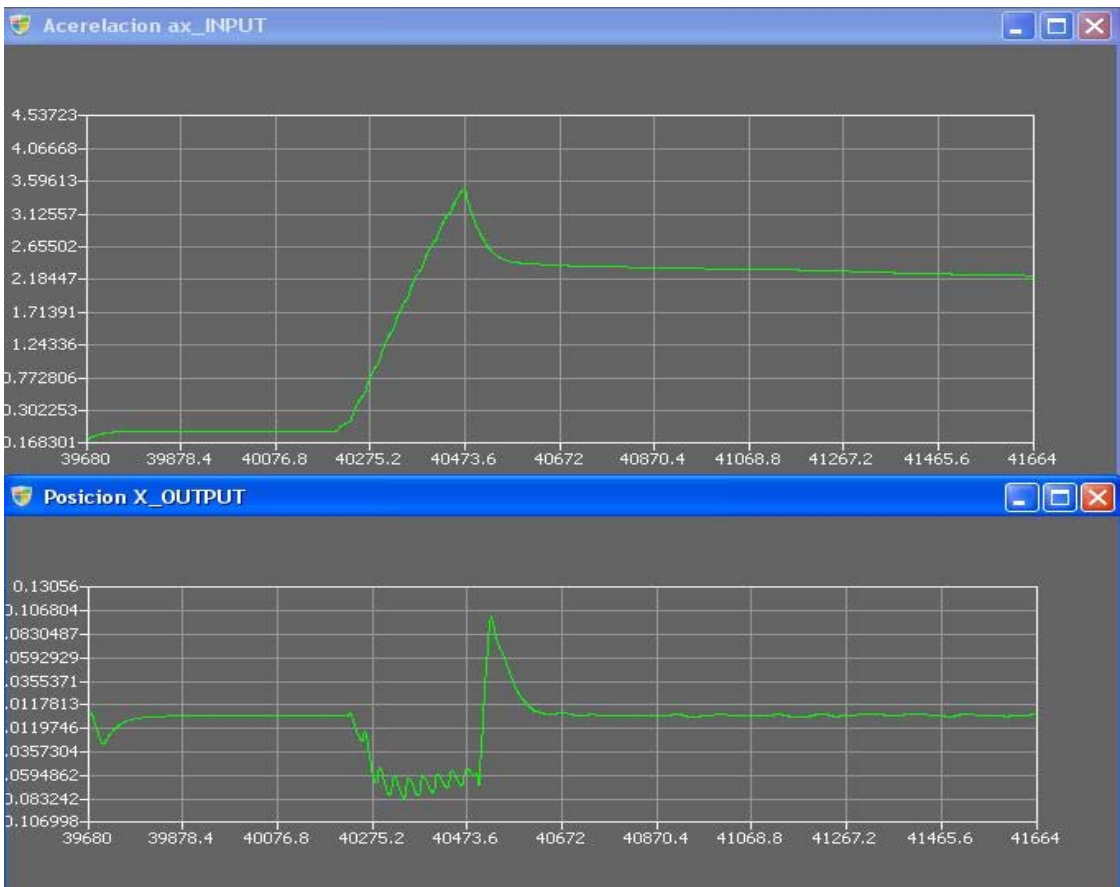


Figura 5-9: Grafica prueba ax y salida pos X (3)

- 4) Hasta este punto se acelera menos pero a los 25 comienza una deceleración que terminará con el vehículo parado y la aceleración lineal en x a cero.

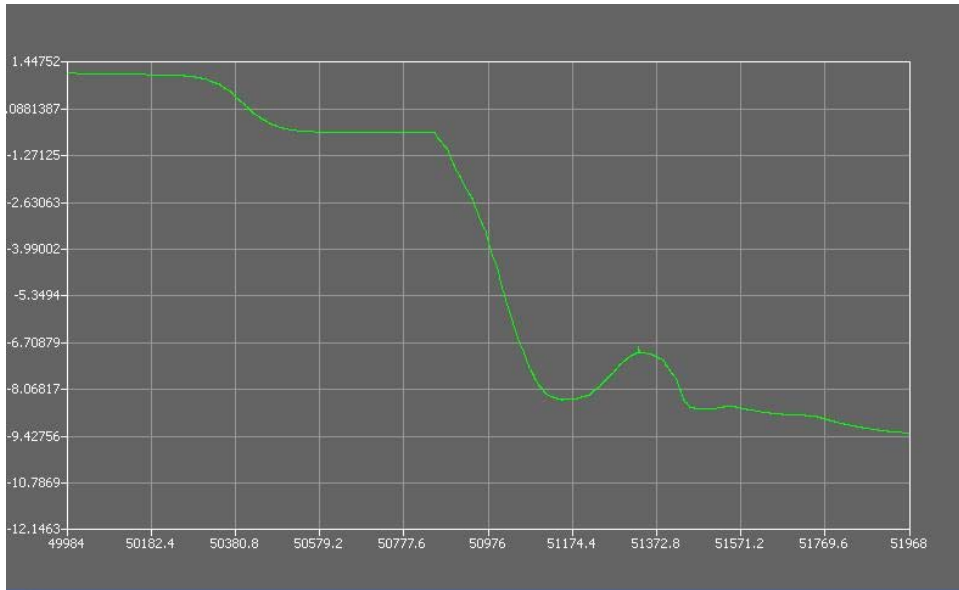


Figura 5-10: Grafica prueba ax (4)

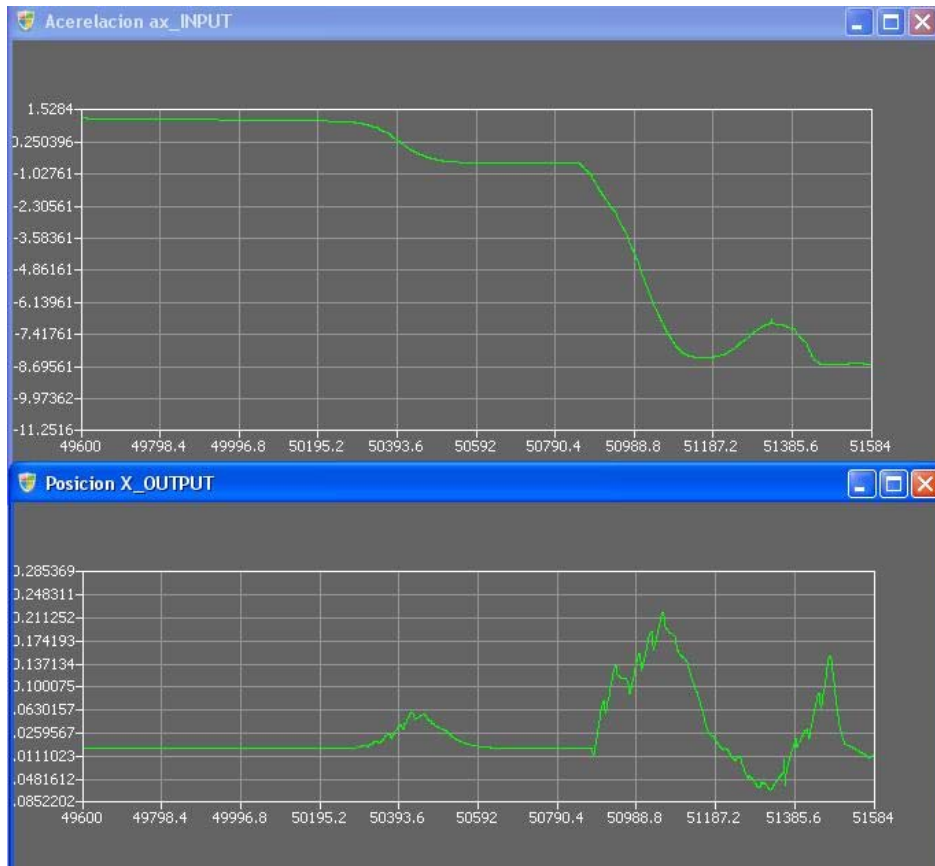


Figura 5-11: Grafica prueba ax y salida pos X (4)

Cómo se puede observar en las gráficas sobre estas líneas el canal referente a la posición de la plataforma cumple su objetivo de funcionamiento. La plataforma siempre vuelve a la posición neutra ($x = 0$) manteniéndose estable a no ser que haya una variación de la aceleración. Si hay una variación de aceleración se pueden ver dos posibles comportamientos:

-Aceleración positiva: la plataforma se mueve hacia atrás. El peso del conductor se traslada hacia atrás del asiento lo que transmite la sensación de aceleración al usuario. Progresivamente la plataforma vuelve a su posición inicial a la espera de nuevas aceleraciones.

-Aceleración negativa: la plataforma se mueve hacia delante. El peso del conductor se desplaza hacia el volante transmitiendo la misma sensación que si el usuario hubiese frenado en un vehículo real. Progresivamente la plataforma vuelve a su posición inicial.

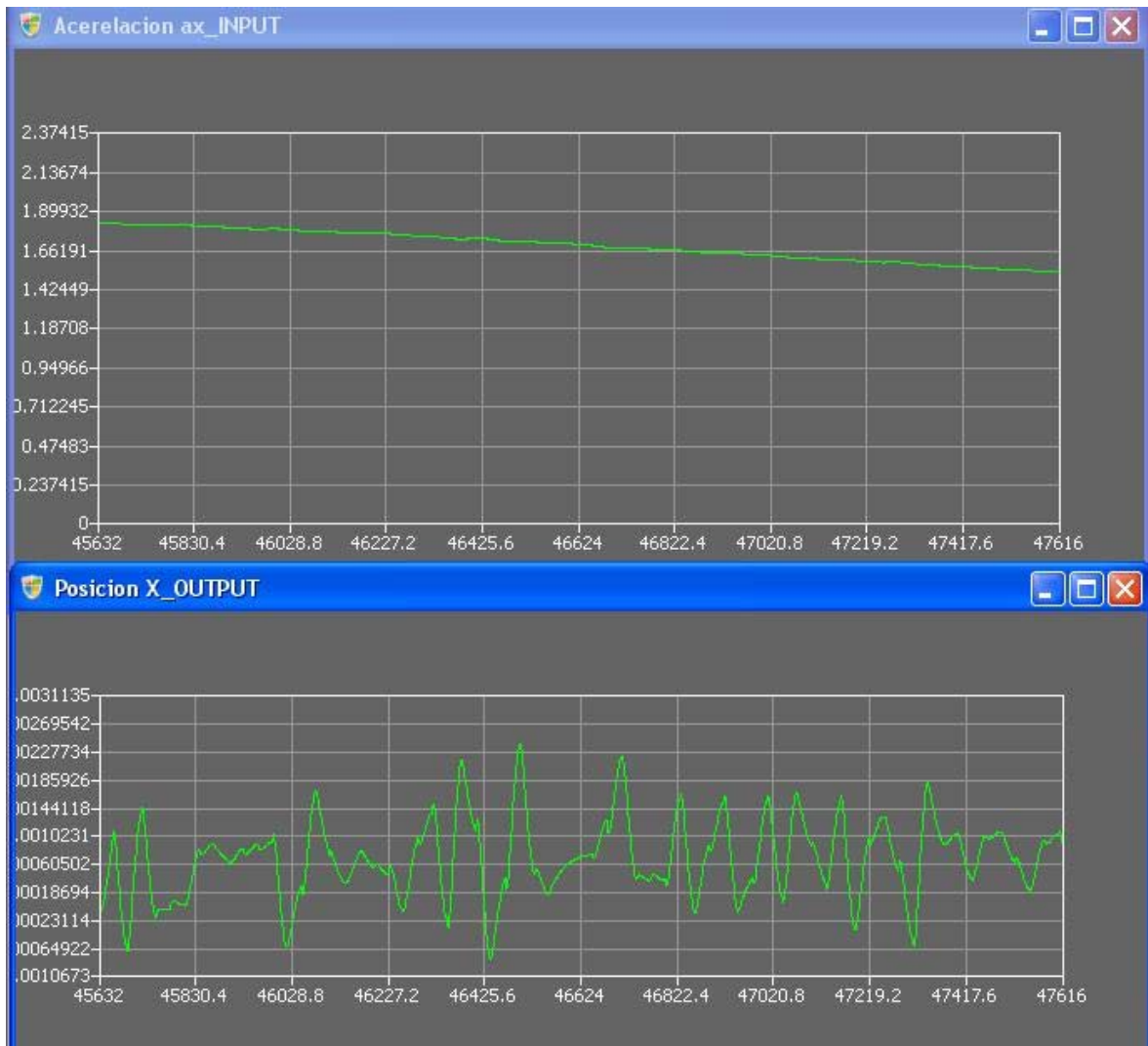


Figura 5-12: Grafica análisis estabilidad

Se puede analizar que la configuración del sistema a pesar de ser eficiente y cumplir su misión cuenta con una cierta inestabilidad. Se puede observar en la figura 5-13 como una ligera deceleración da un resultado con bastante rizado a pesar de que éste tiene máximos y mínimos locales alrededor de cero, el punto inicial de la plataforma. Es un fenómeno que habría que observar en el simulador.

¿Qué ocurre con las otras coordenadas?

Debido a que sufre el mismo proceso visto sobre la coordenada x , las otras coordenadas tienen el mismo comportamiento. Debido a que la base de datos arroja datos de un movimiento sin giros, traza un recorrido recto a lo largo de treinta segundos apenas se debe contemplar movimientos en estas direcciones de la plataforma. Como ejemplo la siguiente imagen.

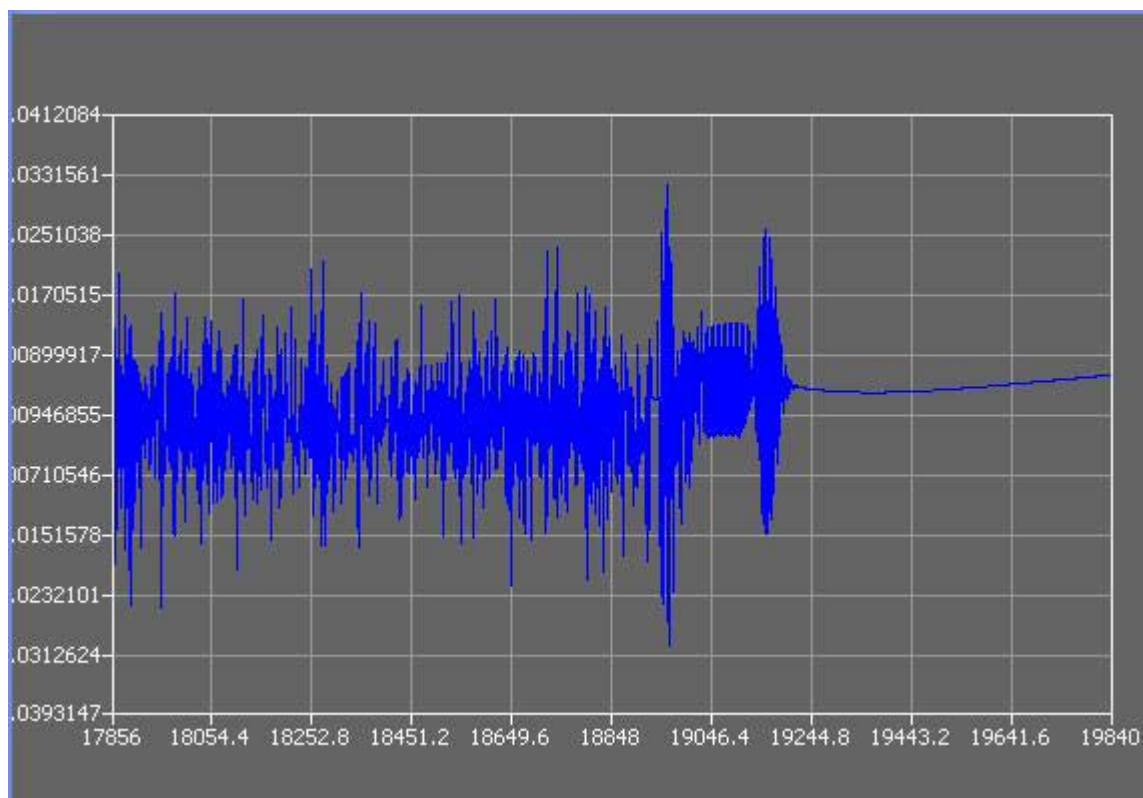


Figura 5-13: Grafica prueba ax

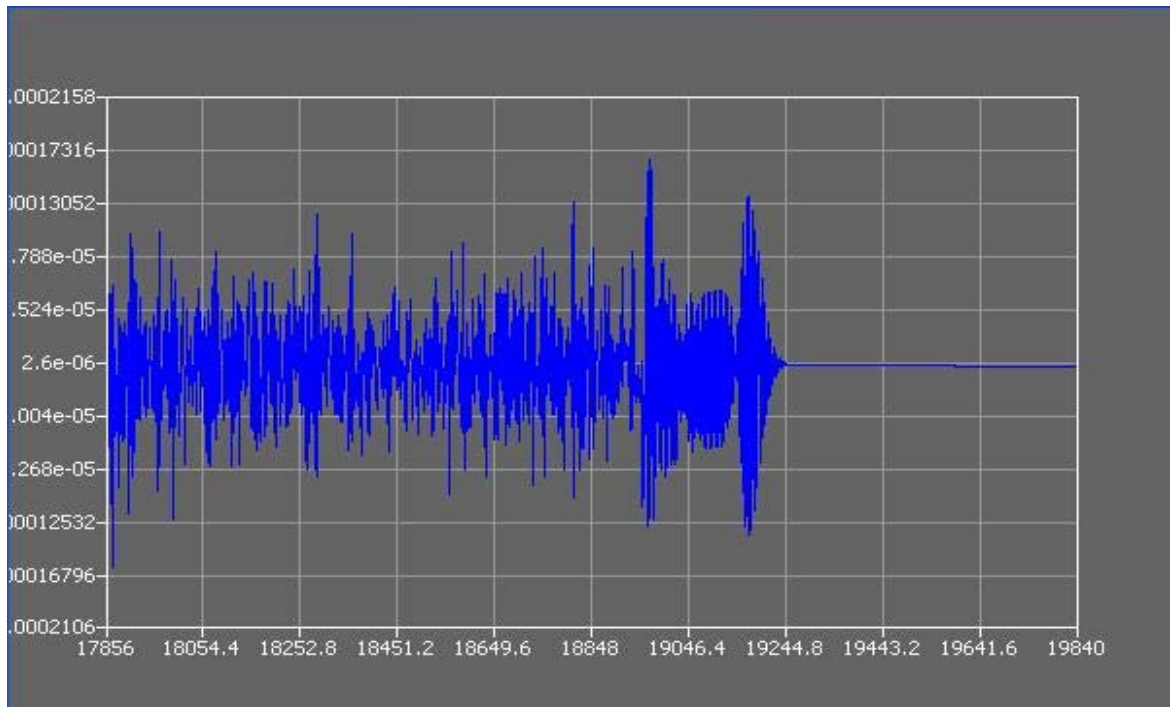


Figura 5-14: Gráfica salida pos Y

5.2.3 Resultados de orientación

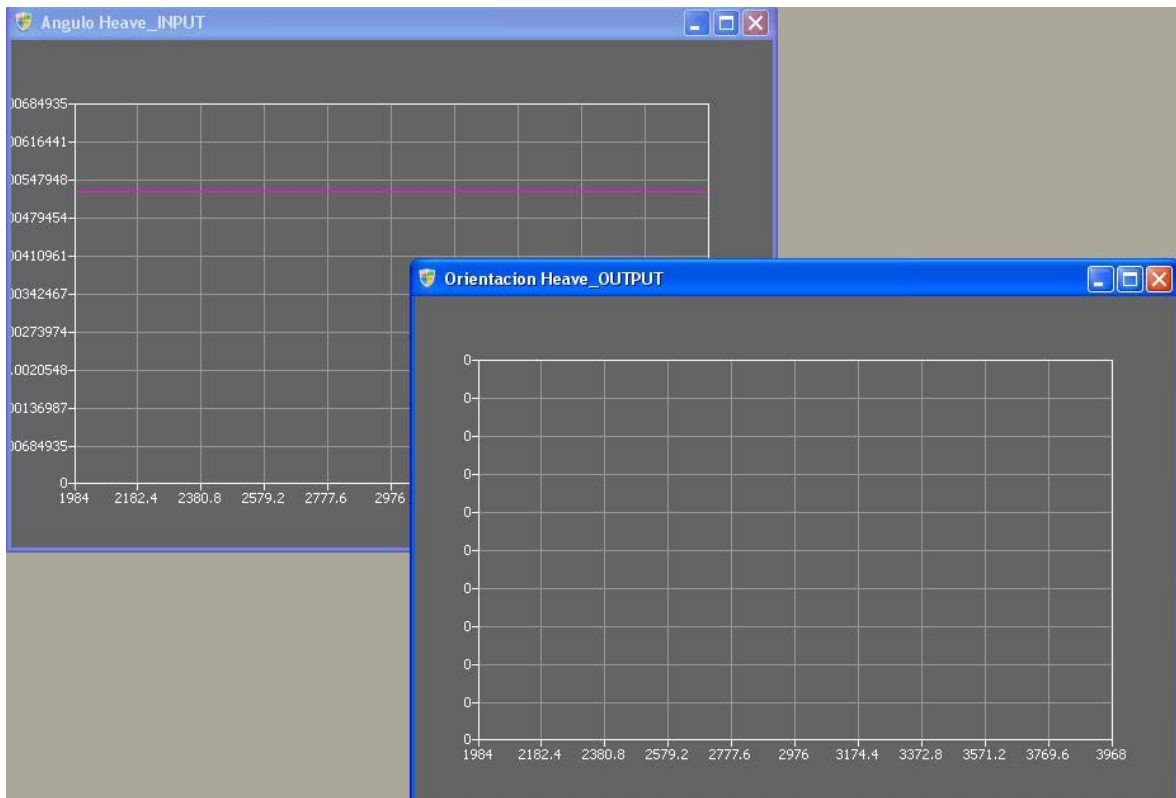


Figura 5-15: Grafica ángulo *heave*

Las pruebas a las que se le ha sometido al sistema de control no contemplan giros ni desplazamientos laterales del automóvil. En la gráfica 5-15 se puede comprobar como el ángulo al ser constante hace que en la salida no haya variaciones de orientación en la plataforma. Confirmando el buen funcionamiento de este canal.

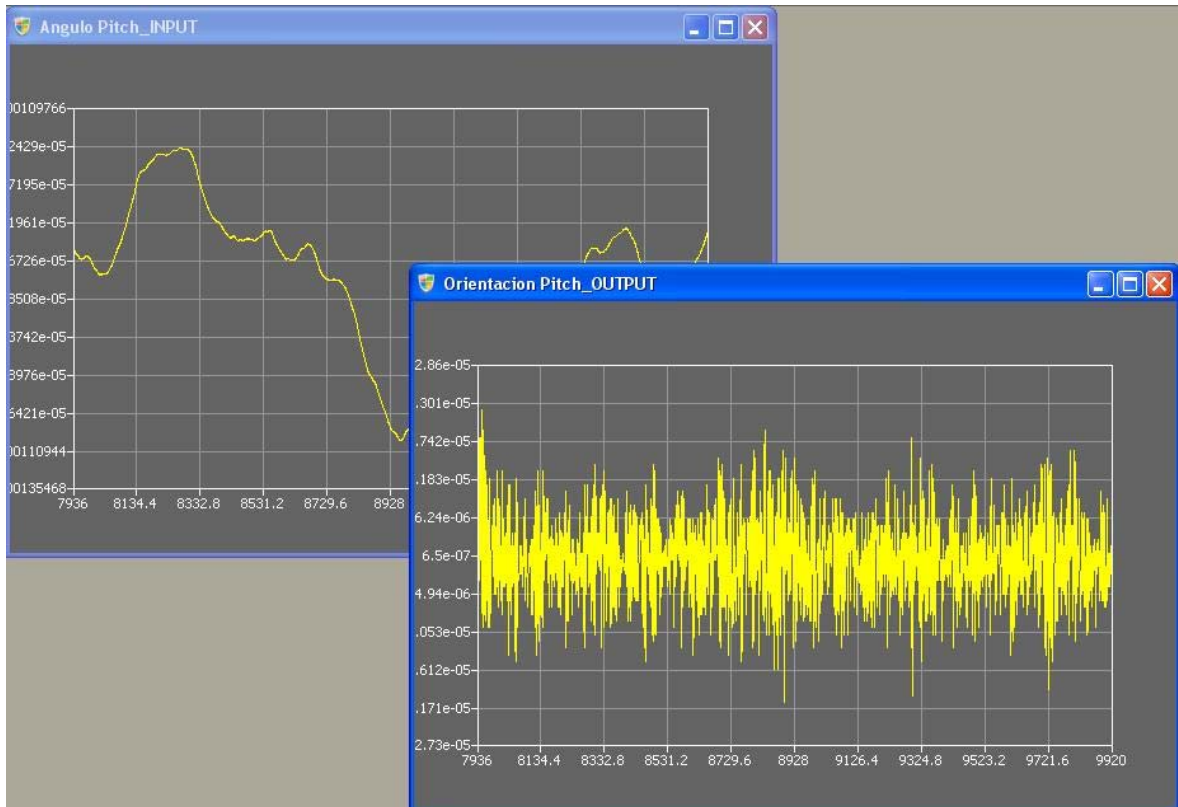


Figura 5-16: Grafica ángulo *pitch*

Las variaciones que se analizan aquí del pitch son mínimas, como mínimos son por tanto los cambios de orientación que son reproducidos por la plataforma. Estas pequeñas variaciones son debidas a que el vehículo no es 100% estable en su movimiento recto, es decir, la parte delantera del vehículo se mueve mínimamente.

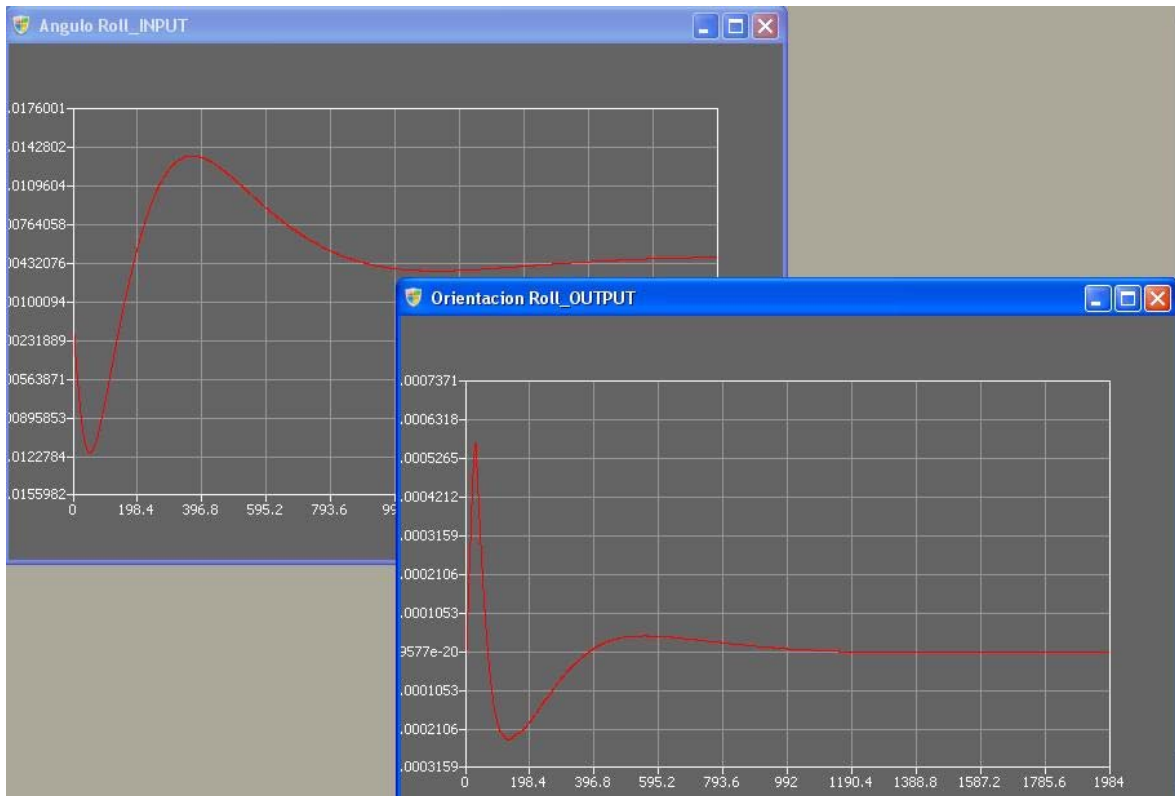


Figura 5-15: Grafica ángulo *roll*

El ángulo *roll* permanece constante a lo largo de la prueba, luego la orientación de la salida hacia la plataforma debe ser cero. Vemos en la gráfica como tras las primeras décimas de segundo de inestabilidad por el encendido del simulador, la plataforma debe situarse en su posición neutra como así sucede.

Las carencias de esta base de datos hacen que la aportación del *tilt-coordination* no sea concluyente. Solo se puede decir que no aporta errores ni desviaciones a la orientación de la misma.

5.3. Pruebas en el simulador

Una vez que los parámetros del sistema de control han sido ajustados, obteniendo unos resultados que encajan con lo esperado y por consiguiente están dentro de los límites que el fabricante especifica para su plataforma, podemos pasar al siguiente nivel de pruebas y ajustes, el simulador.

La planificación del proyecto contemplaba esta parte pero no pudo llevarse a cabo por no estar disponible la opción de usar en esas fechas el simulador de las instalaciones de Indra. Es por tanto ya labor del departamento continuar con esta tarea y la continua actualización del trabajo realizado.



Figura 5-16: Simulador de conducción (II)

6. CONCLUSIONES Y TRABAJO FUTURO

Este proyecto supone la primera piedra de una evolución que se verá reflejada en una experiencia de usuario en el simulador cada vez más próxima a la realidad. Realizando mejoras en el algoritmo, ya sea aplicando nuevas soluciones teóricas como las esbozadas en esta memoria o bien mediante modificaciones experimentales ya que los esquemas de los algoritmos de control del movimiento admiten distintas interpretaciones, filtros y módulos.

La teoría de control de plataformas de movimiento en simuladores continúa evolucionando para aproximarse cada vez más a la realidad. El proyecto realizado es un punto de partida para implementar mejoras en lo que al sistema de control se refiere. El software creado será actualizado, mejorado y potenciado su precisión por el departamento de simulación de Indra.

La literatura en torno al algoritmo clásico propone distintas soluciones de interpretación de cada uno de los canales que lo componen. Añadiendo módulos, eliminando otros, implementando filtros con otra configuración... El punto en el que se encuentra el desarrollo del sistema de control permite que se hagan cambios e ir observando si mejora o no gracias a la visualización de las señales en el sistema de pruebas implementado.

Respecto al sistema de pruebas creado es válido para seguir los cambios que se vayan realizando en el algoritmo original desarrollado. Se podrán añadir funcionalidades nuevas, según vaya desarrollándose el software habrá que cambiar la barra de herramientas para añadir nuevos elementos de ajuste, así como un elemento/botón para retroceder en tiempo de simulación podría ser útil a parte de la posibilidad de acceder al instante deseado que ya está implementado.

A través de las pruebas realizadas en este proyecto se ha puesto de manifiesto cómo la teoría de *washout* cumple su cometido. Su comportamiento es óptimo pero en líneas generales. Demasiado brusco en su comportamiento en ocasiones y eso penalizará las sensaciones recibidas por el conductor en el simulador. Habría que buscar suavizar el comportamiento y evitar dentro de lo posible el rizado a la salida del mismo, a pesar de que sean pequeños valores. En cualquier caso se tendría que evaluar este comportamiento en el simulador antes de realizar cambios en este punto.

En las componentes tangenciales se observa un comportamiento que permite que sea probado el software sin riesgo en el simulador. Pero las variaciones de ángulo son mínimas y con la base de datos propuesta no se pudo realizar un gran ajuste con el sistema de pruebas, habría que observar independientemente estos grados de libertad en el simulador para configurar correctamente los parámetros.

El uso de una base de datos con más posibilidades (frenos y aceleraciones con giros, cambios de nivel...) aportaría una mayor seguridad para certificar el correcto funcionamiento del sistema de control desarrollado.

De la bibliografía y de *papers* consultados pero no incluidos en este proyecto se puede sacar la conclusión general que el futuro de las teorías de control está en la modelización del sistema vestibular y del comportamiento del conductor en el simulador. Existen ya proyectos que van en esa dirección.

Con el tiempo se realizarán cambios en la gestión del movimiento como puede ser una nueva plataforma o añadir un sistema que permita dos nuevos grados de libertad de movimiento independientes. El cliente cuenta ahora con la documentación necesaria para tener una base sobre el funcionamiento de los sistema de control, su teoría, un sistema de pruebas y lo que es más importante, un código propio donde aplicar cambios que permitirán que el simulador se desarrolle al completo, en paralelo todos lo elementos que lo componen.

BIBLIOGRAFÍA

- [1] J.J. Slob, "State-of-the-Art Driving Simulators, a Literature Survey", DCT report Eindhoven University of Technology, August 2008.
- [2] Mike Blundell, Damian Harty, "The multibody systems approach to vehicle dynamics", Elsevier 2004.
- [3] C. Jason Guttridge, "Three Degree-of-Freedom Simulator Motion Cueing Using Classical Washout Filters and Acceleration Feedback", April 2007.
- [4] F. Colombet, M. Dagdelen, G. Reymond, C. Pere, F. Merienne, A. Kemeny, "Motion cueing: what is the impact on the driver's behavior", February 2008
- [5] G. Reymond, A. Heidet, M. Canry, A. Kemeny, "Validation of Renault's dynamic simulator for adaptive cruise control experiments"
- [6] Amira del Rayo, Salvador Galicia y Gabriela Ollyni, "El sistema vestibular: aspectos generales y neurodesarrollo", Universidad de Puebla
- [7] Bruno Daniel Correia Augusto, Universidad técnica de Lisboa, "Motion Cueing in the Chalmers Driving Simulator: An Optimization-Based Control Approach" (2009)
- [8] L. Nehaoua, H. Arioui, S. Espie, H. Mohellebi, "Motion Cueing Algorithms for Small Driving Simulator", IEEE International Conference In Robotics and Automation (ICRA06), Orlando, Florida : United States (2006)
- [9] Han, Lee, "Optimal motion cueing algorithm using the human body model", International Journal (2002)
- [10] Dr Mohamad Kasim Abdul Jalil, "Design and development of 6-DoF motion platform for vehicle driving simulator", Universiti Teknologi Malaysia
- [11] Decker, Dang, Ebert-Uphoff, "Motion Planning for Active Acceleration Compensation ", IEEE (2001)
- [12] Rulph Chassaing, "Digital Signal Processing: Laboratory experiments", Wiley-Interscience (1999)
- [13] Alan V. Oppenheim, Ronald W. Schafer y John R. Buck, "Discrete – time signal processing", Prentice Hall (1999)
- [14] Peter R. Grant, Lloyd D. Reid, "Motion Washout Filter Tuning: Rules and Requirements", University of Toronto, Canada. Journal of Aircraft (1997)
- [15] Bjarne Stroustrup, "C++ programming language", Addison Wesley (1997)
- [16] Jasmin Blanchette, Mark Summerfield, "C++ GUI Programming with Qt 4, Second Edition", Prentice Hall (2008)

Anexo A: LÍMITES DE LA PLATAFORMA

X (MAX)	0,381
X (MIN)	-0,381
Y (MAX)	0,381
Y (MIN)	-0,381
Z (MAX)	0,2286
Z (MIN)	-0,2286
Roll (MAX)	0,50605
Roll (MIN)	-0,50605
Pitch (MAX)	0,57585
Pitch (MIN)	-0,57585
Yaw (MAX)	0,50605
Yaw (MIN)	-0,50605

Tabla A-1: Tilt Rate

Degree of Freedom	Displacement Comb. Motion	Displacement Single DOF	Velocity	Acceleration
Pitch	±22 deg	±21 deg	±30 deg/s	±500 deg/s ²
Roll	±21 deg	±20 deg	±30 deg/s	±500 deg/s ²
Yaw	±23 deg	±22 deg	±40 deg/s	±400 deg/s ²
Heave	±0.18 m (±7.1 in)	±0.18 m (±7.1 in)	±0.30 m/s (±11.8 in/s)	+0.5 G
Surge	±0.27 m (±10.6 in)	±0.25 m (±9.8 in)	±0.50 m/s (±19.7 in/s)	±0.6 G
Sway	±0.26 m (±10.2 in)	±0.25 m (±9.8 in)	±0.50 m/s (±19.7 in/s)	±0.6 G

Tabla A-0-1: Límites de movimiento de la plataforma

Anexo B: CONSTANTE DE GRAVEDAD

El simulador sobre el que se realizan las pruebas se encuentra en Madrid en las instalaciones de Indra en San Fernando de Henares. San Fernando se encuentra a una media de 580 metros de altura sobre el nivel del mar.

La fuerza de gravedad en la superficie de la Tierra no es exactamente igual en todos los sitios. Existen pequeñas variaciones de un lugar a otro. Hay varios factores que intervienen para que así ocurra:

- la distancia hasta el centro del planeta, es decir, su altura;
- de su latitud, ya que la intensidad y la dirección de la aceleración centrífuga varía entre el ecuador y los polos: es máxima en el ecuador y nula en los polos;
- y de la homogeneidad del planeta.

Debido a la rotación de la Tierra, la fuerza o aceleración centrífuga hace disminuir el efecto de la atracción gravitatoria, y la desvía de su dirección original hacia el centro de la Tierra.

El instituto de metrología español determina que al carecer de gravímetro existe una fórmula para determinar de forma aproximada la gravedad local:

$$g_{\phi} = 9,780327 \cdot \left(1 + 0.0053024 \sin^2 \phi - 0.0000058 \sin^2 (2 \cdot \phi)\right) - 3,086 \cdot 10^{-6} \cdot h$$

Donde ϕ es la latitud (decimal) y h es la altura en metros sobre el nivel del mar. La incertidumbre de la fórmula es de 10^{-4} .

En las instalaciones donde se encuentra el simulador la latitud aproximada es 40.425556° y la altura de unos 580 metros. Podemos determinar qué aproximadamente la gravedad será de: 9.7869 m/s^2 .

Puesto que no se trata de un instrumento de alta precisión nos es suficiente la exactitud conseguida.

Anexo C: ¿CÓMO INSTALAR QT?

- 1) Obtener *Qt*. A través de Internet podemos descargar el SDK de *Qt*. Al trabajar en *Windows* instalamos la versión adaptada a esta plataforma (<http://qt.nokia.com/downloads/sdk-windows-cpp>) La versión de la librería con la que trabajamos fue la "2010.02.1"
- 2) Integración con *Visual Studio*. Para poder trabajar con el *Microsoft Visual Studio* es necesario instalar el siguiente paquete (<http://qt.nokia.com/downloads/visual-studio-add-in>)
- 3) Tras la instalación se crea en C:\Archivos de programa una carpeta con el nombre "*Qt*". En su interior encontraremos una carpeta con el nombre de la versión instalada. Se realizará una copia de la misma y se le cambiará el nombre.

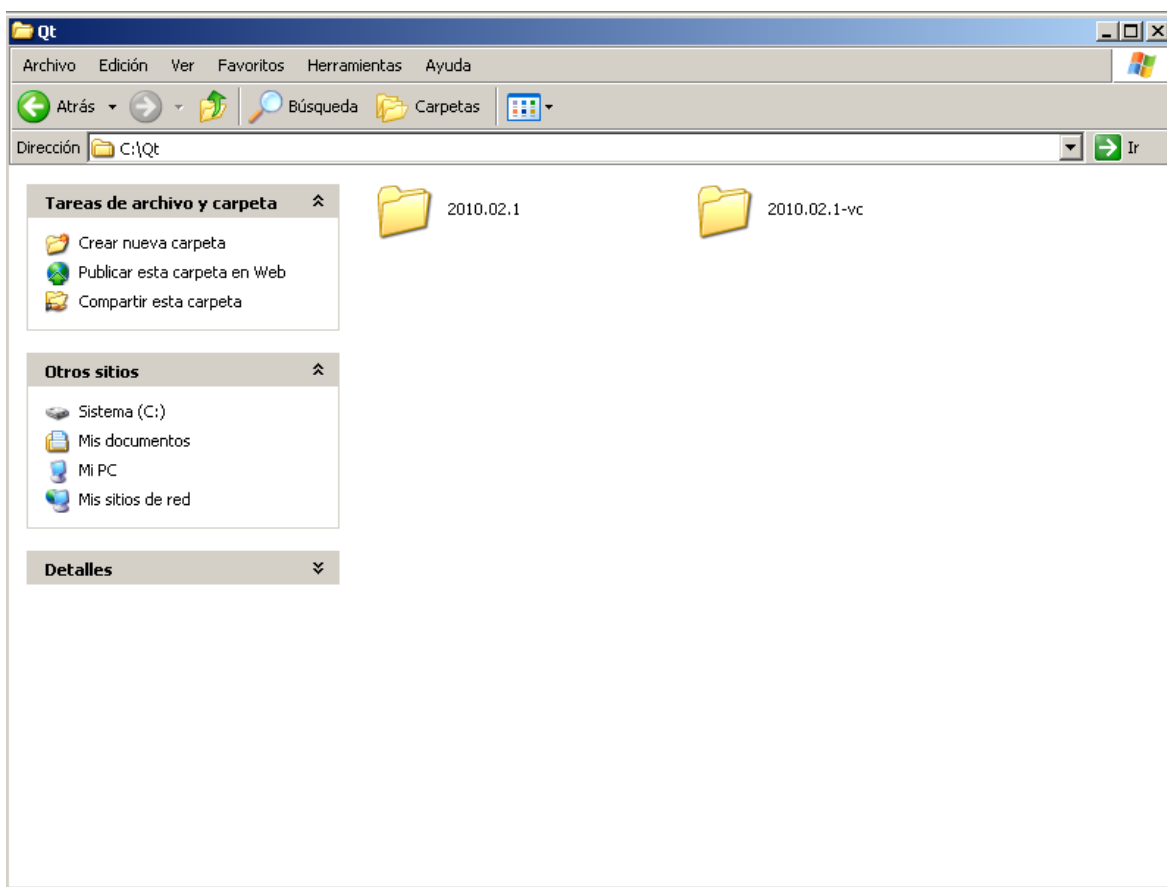
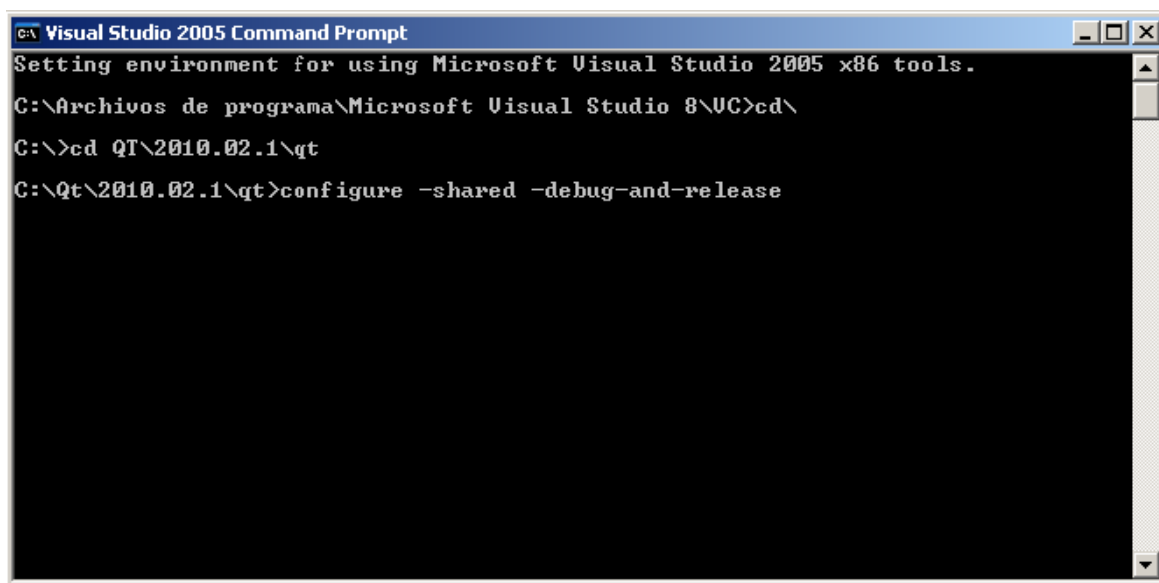


Figura C-1: carpeta Qt

- 4) A continuación dentro de "Visual Studio Tools" ejecutaremos la consola "Visual Studio Command Prompt". Una vez abierta la ventana accedemos a la carpeta C:\Qt\<versión>\qt y ejecutamos la siguiente instrucción > configure -shared -debug-and-release.



```
Visual Studio 2005 Command Prompt
Setting environment for using Microsoft Visual Studio 2005 x86 tools.
C:\Archivos de programa\Microsoft Visual Studio 8\VC>cd\
C:\>cd QT\2010.02.1\qt
C:\Qt\2010.02.1\qt>configure -shared -debug-and-release
```

Figura C-2: Visual Studio 2005 Command Prompt

Se nos preguntará qué versión queremos, teclearemos 'o' de *OpenSource*, y tras aceptar la licencia, comenzará la compilación de *Qt*.

5) Accedemos a Microsoft Visual Studio, ahora encontramos en la barra de menús "Qt". Dentro de "Qt Options" añadiremos la carpeta duplicada que creamos en el paso 3. En este punto ya tendremos concluida el proceso de integración.

A tener en cuenta que cuando se instala una nueva versión de *Qt*, los archivos de configuración del código deben ser modificados para que pueda funcionar el programa.



About Qt

This program uses Qt version 4.6.3.

Qt is a C++ toolkit for cross-platform application development.

Qt provides single-source portability across MS Windows, Mac OS X, Linux, and all major commercial Unix variants. Qt is also available for embedded devices as Qt for Embedded Linux and Qt for Windows CE.

Qt is available under three different licensing options designed to accommodate the needs of our various users.

Qt licensed under our commercial license agreement is appropriate for development of proprietary/commercial software where you do not want to share any source code with third parties or otherwise cannot comply with the terms of the GNU LGPL version 2.1 or GNU GPL version 3.0.

Qt licensed under the GNU LGPL version 2.1 is appropriate for the development of Qt applications (proprietary or open source) provided you can comply with the terms and conditions of the GNU LGPL version 2.1.

Qt licensed under the GNU General Public License version 3.0 is appropriate for the development of Qt applications where you wish to use such applications in combination with software subject to the terms of the GNU GPL version 3.0 or where you are otherwise willing to comply with the terms of the GNU GPL version 3.0.

Please see qt.nokia.com/products/licensing for an overview of Qt licensing.

Copyright (C) 2010 Nokia Corporation and/or its subsidiary(-ies).

Qt is a Nokia product. See qt.nokia.com for more information.

OK

Figura C-3: About Qt

Anexo D: CAN BUS

El CAN bus es un protocolo de comunicaciones serial bus para conectar unidades de control electrónicas, para entornos distribuidos, creado por la firma Bosch. Esta tecnología permite gestionar la comunicación entre múltiples CPUs (nodos)

Cada nodo está preparado para enviar y recibir mensajes pero no simultáneamente. Un mensaje consiste de una ID (para identificar remitente o tipo de mensaje) más los datos. Los bytes son transmitidos en serie por el bus codificados en NRZ.

Los dispositivos típicos que se interconectan, haciendo uso de un controlador CAN, en este tipo de redes son sensores, actuadores, etc.

Cada nodo consta de:

*Procesador: para descifrar el tipo de mensaje recibido y que mensaje se quiere enviar.

*Controlador CAN: cuando recibe, almacenamiento de bits antes de pasárselos al procesador y en el envío tiene guardado el mensaje antes de enviarlo por el bus. El controlador también adapta la señal al bus.

Anexo E: VÍDEOS

Esta memoria se acompaña de un CD con los siguientes videos:

- Presentación del simulador
- El futuro de los simuladores
- Funcionamiento programa de pruebas.

PRESUPUESTO

1) Ejecución Material

- Compra de ordenador personal (Software incluido)..... 2.000 €
- Alquiler de impresora láser durante 6 meses 50 €
- Material de oficina 150 €
- Total de ejecución material 2.200 €

2) Gastos generales

- 15 % sobre Ejecución Material 330 €

3) Honorarios Proyecto

- 700 horas a 30 € / hora 21000 €

4) Material fungible

- Gastos de impresión 80 €
- Encuadernación 10 €

5) Subtotal del presupuesto

- Subtotal Presupuesto 23620 €

6) I.V.A. aplicable

- 18% Subtotal Presupuesto 4251,60 €

7) Total presupuesto

- Total Presupuesto 27871,60 €

Madrid, Noviembre de 2010

El Ingeniero Jefe de Proyecto

Fdo.: Sergio Orzáez Sánchez
Ingeniero Superior de Telecomunicación

PLIEGO DE CONDICIONES

Este documento contiene las condiciones legales que guiarán la realización, en este proyecto, de un desarrollo de rutinas de filtrado para un sistema de movimiento de 6 grados de libertad. En lo que sigue, se supondrá que el proyecto ha sido encargado por una empresa cliente a una empresa consultora con la finalidad de realizar dicho sistema. Dicha empresa ha debido desarrollar una línea de investigación con objeto de elaborar el proyecto. Esta línea de investigación, junto con el posterior desarrollo de los programas está amparada por las condiciones particulares del siguiente pliego.

Supuesto que la utilización industrial de los métodos recogidos en el presente proyecto ha sido decidida por parte de la empresa cliente o de otras, la obra a realizar se regulará por las siguientes:

Condiciones generales

1. La modalidad de contratación será el concurso. La adjudicación se hará, por tanto, a la proposición más favorable sin atender exclusivamente al valor económico, dependiendo de las mayores garantías ofrecidas. La empresa que somete el proyecto a concurso se reserva el derecho a declararlo desierto.

2. El montaje y mecanización completa de los equipos que intervengan será realizado totalmente por la empresa licitadora.

3. En la oferta, se hará constar el precio total por el que se compromete a realizar la obra y el tanto por ciento de baja que supone este precio en relación con un importe límite si este se hubiera fijado.

4. La obra se realizará bajo la dirección técnica de un Ingeniero Superior de Telecomunicación, auxiliado por el número de Ingenieros Técnicos y Programadores que se estime preciso para el desarrollo de la misma.

5. Aparte del Ingeniero Director, el contratista tendrá derecho a contratar al resto del personal, pudiendo ceder esta prerrogativa a favor del Ingeniero Director, quien no estará obligado a aceptarla.

6. El contratista tiene derecho a sacar copias a su costa de los planos, pliego de condiciones y presupuestos. El Ingeniero autor del proyecto autorizará con su firma las copias solicitadas por el contratista después de confrontarlas.

7. Se abonará al contratista la obra que realmente ejecute con sujeción al proyecto que sirvió de base para la contratación, a las modificaciones autorizadas por

la superioridad o a las órdenes que con arreglo a sus facultades le hayan comunicado por escrito al Ingeniero Director de obras siempre que dicha obra se haya ajustado a los preceptos de los pliegos de condiciones, con arreglo a los cuales, se harán las modificaciones y la valoración de las diversas unidades sin que el importe total pueda exceder de los presupuestos aprobados. Por consiguiente, el número de unidades que se consignan en el proyecto o en el presupuesto, no podrá servirle de fundamento para entablar reclamaciones de ninguna clase, salvo en los casos de rescisión.

8. Tanto en las certificaciones de obras como en la liquidación final, se abonarán los trabajos realizados por el contratista a los precios de ejecución material que figuran en el presupuesto para cada unidad de la obra.

9. Si excepcionalmente se hubiera ejecutado algún trabajo que no se ajustase a las condiciones de la contrata pero que sin embargo es admisible a juicio del Ingeniero Director de obras, se dará conocimiento a la Dirección, proponiendo a la vez la rebaja de precios que el Ingeniero estime justa y si la Dirección resolviera aceptar la obra, quedará el contratista obligado a conformarse con la rebaja acordada.

10. Cuando se juzgue necesario emplear materiales o ejecutar obras que no figuren en el presupuesto de la contrata, se evaluará su importe a los precios asignados a otras obras o materiales análogos si los hubiere y cuando no, se discutirán entre el Ingeniero Director y el contratista, sometiéndolos a la aprobación de la Dirección. Los nuevos precios convenidos por uno u otro procedimiento, se sujetarán siempre al establecido en el punto anterior.

11. Cuando el contratista, con autorización del Ingeniero Director de obras, emplee materiales de calidad más elevada o de mayores dimensiones de lo estipulado en el proyecto, o sustituya una clase de fabricación por otra que tenga asignado mayor precio o ejecute con mayores dimensiones cualquier otra parte de las obras, o en general, introduzca en ellas cualquier modificación que sea beneficiosa a juicio del Ingeniero Director de obras, no tendrá derecho sin embargo, sino a lo que le correspondería si hubiera realizado la obra con estricta sujeción a lo proyectado y contratado.

12. Las cantidades calculadas para obras accesorias, aunque figuren por partidaalzada en el presupuesto final (general), no serán abonadas sino a los precios de la contrata, según las condiciones de la misma y los proyectos particulares que para ellas se formen, o en su defecto, por lo que resulte de su medición final.

13. El contratista queda obligado a abonar al Ingeniero autor del proyecto y director de obras así como a los Ingenieros Técnicos, el importe de sus respectivos honorarios facultativos por formación del proyecto, dirección técnica y administración en su caso, con arreglo a las tarifas y honorarios vigentes.

14. Concluida la ejecución de la obra, será reconocida por el Ingeniero Director que a tal efecto designe la empresa.

15. La garantía definitiva será del 4% del presupuesto y la provisional del 2%.

16. La forma de pago será por certificaciones mensuales de la obra ejecutada, de acuerdo con los precios del presupuesto, deducida la baja si la hubiera.

17. La fecha de comienzo de las obras será a partir de los 15 días naturales del replanteo oficial de las mismas y la definitiva, al año de haber ejecutado la provisional, procediéndose si no existe reclamación alguna, a la reclamación de la fianza.

18. Si el contratista al efectuar el replanteo, observase algún error en el proyecto, deberá comunicarlo en el plazo de quince días al Ingeniero Director de obras, pues transcurrido ese plazo será responsable de la exactitud del proyecto.

19. El contratista está obligado a designar una persona responsable que se entenderá con el Ingeniero Director de obras, o con el delegado que éste designe, para todo relacionado con ella. Al ser el Ingeniero Director de obras el que interpreta el proyecto, el contratista deberá consultarle cualquier duda que surja en su realización.

20. Durante la realización de la obra, se girarán visitas de inspección por personal facultativo de la empresa cliente, para hacer las comprobaciones que se crean oportunas. Es obligación del contratista, la conservación de la obra ya ejecutada hasta la recepción de la misma, por lo que el deterioro parcial o total de ella, aunque sea por agentes atmosféricos u otras causas, deberá ser reparado o reconstruido por su cuenta.

21. El contratista, deberá realizar la obra en el plazo mencionado a partir de la fecha del contrato, incurriendo en multa, por retraso de la ejecución siempre que éste no sea debido a causas de fuerza mayor. A la terminación de la obra, se hará una recepción provisional previo reconocimiento y examen por la dirección técnica, el depositario de efectos, el interventor y el jefe de servicio o un representante, estampando su conformidad el contratista.

22. Hecha la recepción provisional, se certificará al contratista el resto de la obra, reservándose la administración el importe de los gastos de conservación de la misma hasta su recepción definitiva y la fianza durante el tiempo señalado como plazo de garantía. La recepción definitiva se hará en las mismas condiciones que la provisional, extendiéndose el acta correspondiente. El Director Técnico propondrá a la Junta Económica la devolución de la fianza al contratista de acuerdo con las condiciones económicas legales establecidas.

23. Las tarifas para la determinación de honorarios, reguladas por orden de la Presidencia del Gobierno el 19 de Octubre de 1961, se aplicarán sobre el denominado en la actualidad "Presupuesto de Ejecución de Contrata" y anteriormente llamado "Presupuesto de Ejecución Material" que hoy designa otro concepto.

Condiciones particulares

La empresa consultora, que ha desarrollado el presente proyecto, lo entregará a la empresa cliente bajo las condiciones generales ya formuladas, debiendo añadirse las siguientes condiciones particulares:

1. La propiedad intelectual de los procesos descritos y analizados en el presente trabajo, pertenece por entero a la empresa consultora representada por el Ingeniero Director del Proyecto.

2. La empresa consultora se reserva el derecho a la utilización total o parcial de los resultados de la investigación realizada para desarrollar el siguiente proyecto, bien para su publicación o bien para su uso en trabajos o proyectos posteriores, para la misma empresa cliente o para otra.

3. Cualquier tipo de reproducción aparte de las reseñadas en las condiciones generales, bien sea para uso particular de la empresa cliente, o para cualquier otra aplicación, contará con autorización expresa y por escrito del Ingeniero Director del Proyecto, que actuará en representación de la empresa consultora.

4. En la autorización se ha de hacer constar la aplicación a que se destinan sus reproducciones así como su cantidad.

5. En todas las reproducciones se indicará su procedencia, explicitando el nombre del proyecto, nombre del Ingeniero Director y de la empresa consultora.

6. Si el proyecto pasa la etapa de desarrollo, cualquier modificación que se realice sobre él, deberá ser notificada al Ingeniero Director del Proyecto y a criterio de éste, la empresa consultora decidirá aceptar o no la modificación propuesta.

7. Si la modificación se acepta, la empresa consultora se hará responsable al mismo nivel que el proyecto inicial del que resulta el añadirla.

8. Si la modificación no es aceptada, por el contrario, la empresa consultora declinará toda responsabilidad que se derive de la aplicación o influencia de la misma.

9. Si la empresa cliente decide desarrollar industrialmente uno o varios productos en los que resulte parcial o totalmente aplicable el estudio de este proyecto, deberá comunicarlo a la empresa consultora.

10. La empresa consultora no se responsabiliza de los efectos laterales que se puedan producir en el momento en que se utilice la herramienta objeto del presente proyecto para la realización de otras aplicaciones.

11. La empresa consultora tendrá prioridad respecto a otras en la elaboración de los proyectos auxiliares que fuese necesario desarrollar para dicha aplicación industrial, siempre que no haga explícita renuncia a este hecho. En este caso, deberá autorizar expresamente los proyectos presentados por otros.

12. El Ingeniero Director del presente proyecto, será el responsable de la dirección de la aplicación industrial siempre que la empresa consultora lo estime oportuno. En caso contrario, la persona designada deberá contar con la autorización del mismo, quien delegará en él las responsabilidades que ostente.