



## XVI CONGRESO INTERNACIONAL DE INGENIERÍA GRÁFICA

---

### CREACIÓN DE ENTORNOS 3D PARA UN SIMULADOR DE CONDUCCIÓN DE AUTOMÓVILES

MORER CAMO, Paz (1); NAYA VILLAVERDE, Miguel Angel (1);  
MONZÓN GÓMEZ, Luis (1)

<sup>(1)</sup>Universidad de A Coruña, España  
Escuela Politécnica Superior  
[pmorer@udc.es](mailto:pmorer@udc.es)  
[minaya@cdf.udc.es](mailto:minaya@cdf.udc.es)

#### RESUMEN

El objeto de este trabajo se centra en el desarrollo de una herramienta que permita cargar diversos entornos 3D, con cierto grado de realismo, y de los que se pueda obtener información geométrica del terreno, de manera que, en cada punto, se pueda saber con precisión las características que lo determinan. Esta información será utilizada para resolver las diferentes posiciones del vehículo, partiendo de unas fuerzas iniciales, velocidad, ... y obteniendo una respuesta en tiempo real. Financiado por la CICYT en el periodo 2001-03.

**Palabras clave:** Visualización de entornos 3D. Tiempo Real. Sistemas multicuerpo.

#### ABSTRACT

This job has the purpose of developing a tool that allows loading different 3D worlds, with information about terrains, textures, culture, and vegetation; so we can know with certain precision their geometric characteristics, place a car in this geometry; and drive it on the terrain with extern hardware. This job was financed by CICYT during the period 2001-2003.

**Key words:** Visualization of 3D world. Real Time. Multibody Systems.

## 1. Introducción

Este trabajo se enmarca dentro del Proyecto “Un entorno unificado de simulación para la dinámica en tiempo real de sistemas multicuerpo con análisis tensional y control” financiado por la CICYT para el trienio 2001-2003.

Su objetivo es la creación de entornos 3D de propósito general, que permita visualizar diversos terrenos, a partir de datos geoespaciales y distintos modelos de vehículos, sin necesidad de tener que volver a reprogramar el código adecuándolo a los nuevos datos. A su vez se pretende controlar el vehículo mediante un hardware externo que permita “conducir” por ese terreno.

Se parte de un modelo de simulación del control del automóvil, correspondiente a un vehículo experimental del Laboratorio de Ingeniería Mecánica. La programación se ha desarrollado sobre las librerías gráficas del Virtual Terrain Project y OpenScene Graph, en un entorno de Microsoft Visual C++.

Otros trabajos realizados en esta línea y pueden verse en [1]

## 2. Herramientas Utilizadas

Existen numerosas herramientas, que permiten la generación de entornos 3D, la mayoría de ellas están basadas sobre las librerías gráficas de OpenGL, actualmente un estándar en gráficos, y soportada por la mayoría de las tarjetas gráficas y los diferentes sistemas operativos. Entre ellas podemos citar: “Open Inventor”, “Open Performer”, “OpenScene Graph”, “The Mesa 3D Graphics Library”, entre otros.

Entre las diferentes opciones se ha optado por VTP “Virtual Terrain Project” [2], al tratarse de un conjunto de librerías gráficas de alto nivel, desarrolladas con el objetivo de facilitar la creación de nuevas herramientas que permitan la representación de cualquier parte del mundo real, en formato 3D digital e interactivo, de una manera sencilla y rápida.

“Virtual Terrain Project” supone una puesta en común y conjunta del CAD, GIS, Simulación, Topografía. Los campos de aplicación varían desde el turismo virtual, los impactos de la construcción en determinados entornos, la educación en temas geográficos, la visualización de la climatología, juegos, etc.

El código es abierto y de libre distribución. Ofrece un conjunto de herramientas de programación “VTP ToolBox” y una aplicación abierta “VTP Enviro” para trabajar con mapas de terrenos.

La siguiente figura muestra la jerarquía establecida de las herramientas desarrolladas. Las aplicaciones Enviro y VTBuilder se desarrollan sobre la librería vtlib y las librerías wxWindows, estas últimas permitirán desarrollar el código de programación referente al interface y dialogo del usuario con la aplicación.



A la par que se ha desarrollado la aplicación sobre estas herramientas, se han utilizado programas de Cad comerciales para la modelización de elementos, como es el caso de los modelos de automóviles, en el que se distinguirán las partes fundamentales para el propósito que nos concierne: como son el chasis, carrocería interior y exterior y las ruedas. Estos programas de Cad se refieren al 3D-Studio, y el Rhinoceros, este último utilizado como intercambiador de ficheros.

Por último, la plataforma sobre la que se ha desarrollado este trabajo es el entorno de programación Visual C++ y Visual Fortran, debido a que la programación correspondiente a la parte gráfica se realiza en C++, mientras que el desarrollo correspondiente al cálculo de la dinámica se ha realizado en Fortran.

### 3. Creación del Entorno 3D de la Simulación

Los datos del terreno se presentan en varios archivos de distintos tipos de formatos. Un tipo de archivos contendrá la información sobre la geometría del terreno, este tipo de archivos los distinguiremos por la extensión .bt. Este formato se comenzó a diseñar en 1997 ante la precariedad existente a la hora de encontrar formatos para el trabajo con mallas de generación de terrenos. Los formatos estándar existentes entonces .dem (USGS Digital Elevation Model) [3], tan sólo eran adecuados para terrenos de tamaño estándar. Se trata de un formato simple, compacto, y fácil de leer y escribir, flexible en lo que se refiere al tamaño de terrenos y los sistemas de coordenadas a emplear.

Los terrenos que requieren una definición más compleja tienen asociados un fichero de proyecciones de extensión .prj, de mismo nombre y situados en el mismo directorio. Estos ficheros contienen una descripción del sistema de coordenadas empleado en la proyección de los datos geoespaciales.

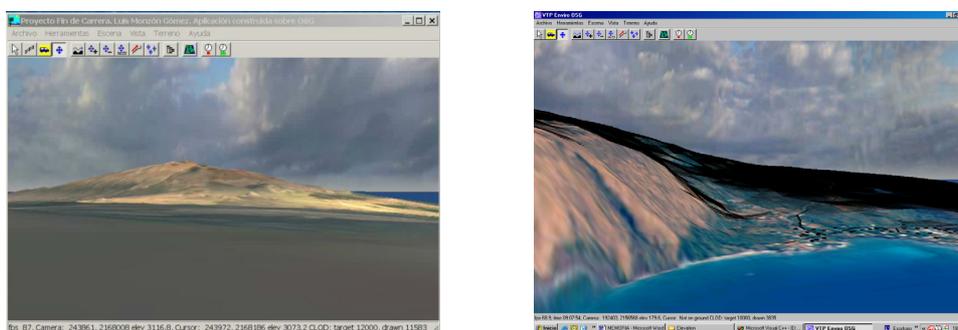


Figura 2: “Ventanas de visualización de un entorno”

El resto de los archivos deberán almacenar la información sobre la textura del terreno, en formatos .jpg, .bmp, o sobre las imágenes relativas al cielo, o información sobre carreteras, vegetación, edificios, ...

Al comenzar la aplicación se selecciona un fichero de extensión .ini, el cual recoge la información concreta de la ubicación de los ficheros .bt, .prj, .jpg, .bmp, el

nivel de detalle, etc. En el conjunto de la programación podremos distinguir tres grandes bloques en los que se podrán agrupar las diferentes rutinas:



Figura 4: "Distribución del source en bloques"

Cabe destacar el carácter modular que tiene este programa de simulación, lo cual le otorga una gran capacidad para la ampliación de funcionalidades en un futuro próximo. En el bloque que hemos denominado "renderizado", se agrupan las capacidades del programa para establecer un diálogo con el usuario, tanto al inicio de la aplicación como a lo largo de su ejecución. Estos diálogos se llevan a cabo con las librerías wxWindows [4], mediante ventanas, menús desplegables, lectura de eventos del ratón, o mediante teclado etc. Para la lectura de los datos introducidos con el volante, se han desarrollado sus propias rutinas, siendo necesarios los drivers específicos del hardware utilizado.

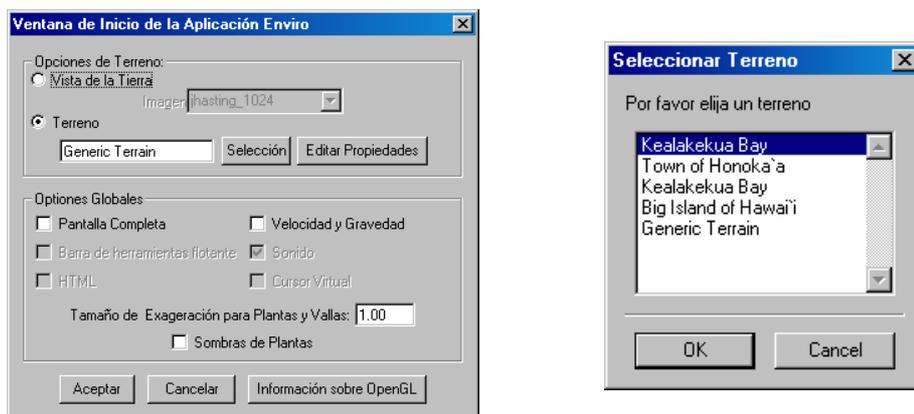


Figura 5: "Ventana de diálogo de inicio"

#### 4. Conexión del Entorno 3D con la Simulación Dinámica del Automóvil

Creado el entorno y abierta la aplicación, es necesario situar en ese entorno el automóvil, punto que se lleva a cabo de una manera sencilla, mediante un simple

“clicado” con el ratón sobre el terreno. El hecho de “picar” un punto nos permite situar el coche respecto a un origen relativo de coordenadas, sin necesidad de saber con anterioridad su situación inicial. Esta opción da una gran libertad, tanto de elección de entornos, como de modelos de automóviles, y de puntos de partida de la simulación.

Situado el automóvil, es necesaria una conexión con el módulo que realiza el cálculo de la dinámica del vehículo. En el primer instante, una vez elegido mediante el ratón la posición de la que partirá el vehículo, el modelo calcula la posición, velocidad y aceleración inicial del coche teniendo en cuenta el terreno sobre el que está colocado y las fuerzas que actúan sobre él.

El vehículo simulado corresponde a un prototipo construido en el Laboratorio de Ingeniería Mecánica, formado por un chasis de tubo de acero, y en el que el resto de los elementos (motor, suspensiones, dirección, ruedas, etc.) han sido adaptados a partir de componentes de vehículos comerciales. La simulación es capaz de representar en tiempo real las características no lineales de los elementos del prototipo. Para las ecuaciones del movimiento se ha utilizado una formulación de Lagrange aumentado de índice 3, siguiendo los algoritmos desarrollados en el Laboratorio [4-5]. El modelo del sistema mecánico resultante consta de 44 puntos, 7 vectores, 1 ángulo y 5 distancias, lo que supone un sistema con 159 variables que se resuelve con un paso de tiempo de 0,01 s. El par del motor se calcula a partir de las curvas de par del motor real, el contacto neumático suelo se obtiene mediante la “Formula Mágica” de Pacejka [6] a partir de los coeficientes suministrados por el fabricante y las fuerzas de frenado se han calculado teniendo en cuenta que el vehículo está dotado de frenos de disco en las cuatro ruedas.

El siguiente gráfico muestra el diagrama del proceso a seguir de la simulación. En él se pueden ver tres funciones importantes, que son las puertas de entrada y salida de la simulación gráfica con las rutinas de la simulación dinámica, que son las rutinas INI, CALCULO, y SUELO. La rutina INI se emplea para obtener la posición, la velocidad y la aceleración iniciales del vehículo. La rutina CALCULO se encarga de calcular el movimiento real del vehículo (posición, velocidad y aceleración de todos los puntos) y las fuerzas internas en algunos elementos como los amortiguadores, para cada paso de tiempo, teniendo en cuenta las fuerzas exteriores que actúan sobre él, y los diferentes eventos introduce el conductor.

# APLICACIÓN DEL SIMULADOR

## DIAGRAMA DEL PROCESO DE SIMULACIÓN

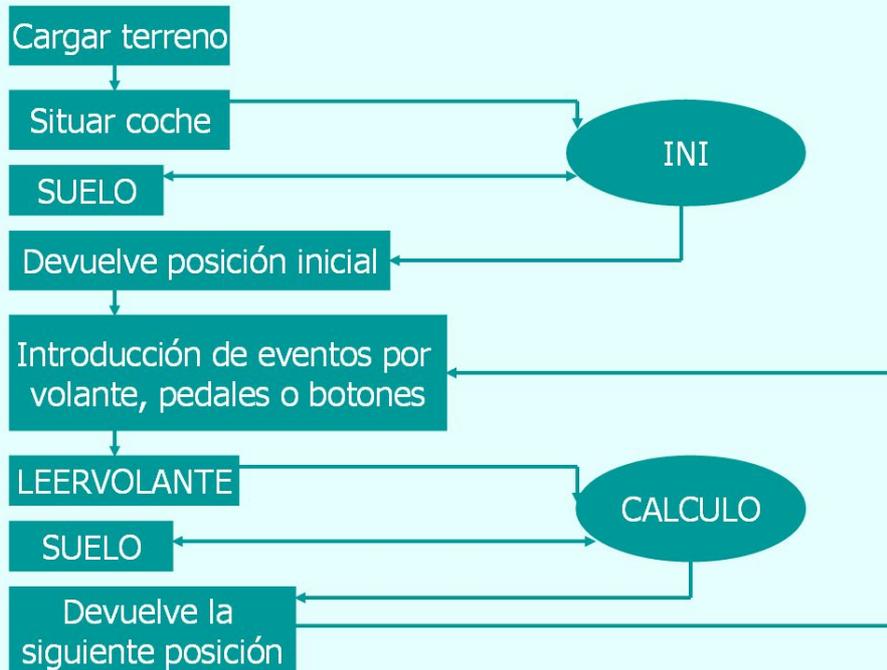
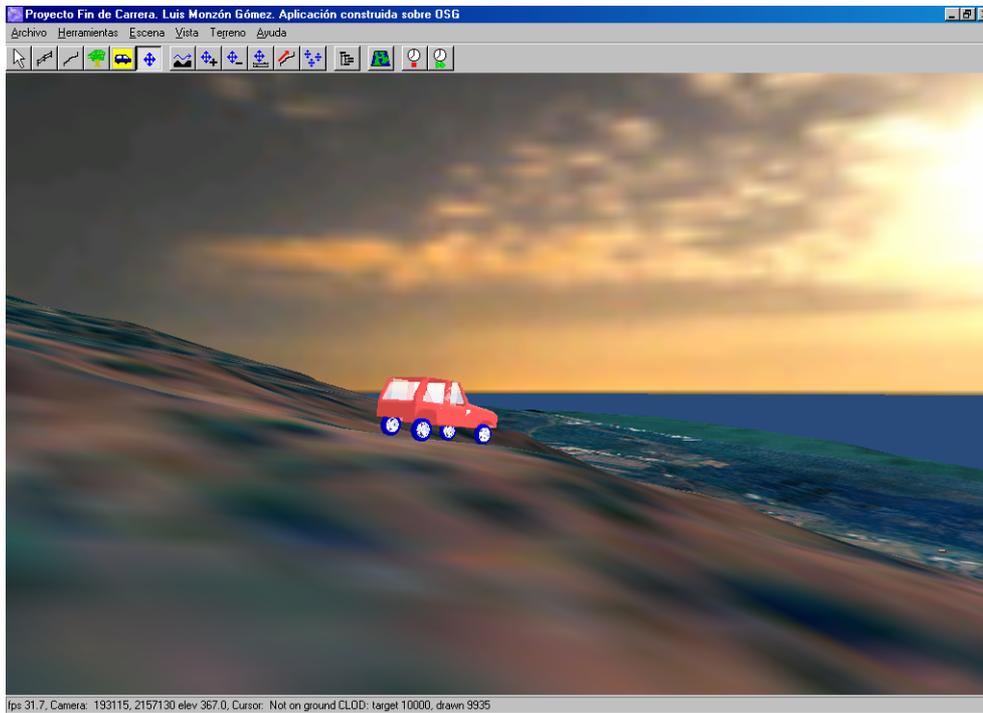


Figura 6. "Diagrama del proceso de simulación"

Un aspecto muy importante es la determinación de la altura del terreno sobre el que se apoya el vehículo. Esta información es suministrada por la subrutina SUELO. Esta función es llamada desde Fortran a C++. La función en Fortran que se ocupa del cálculo, envía una posición conocida (x,y), posición leída por la subrutina SUELO la cual devuelve, para ese x e y, la altitud y la normal al terreno en ese punto. Datos necesarios para poder proseguir el proceso de cálculo. La obtención de la altitud se realiza a partir de la información del mallado del terreno. El mallado es dividido en un número dado de columnas (Este-Oeste) y filas (Norte-Sur), de una extensión definida. El primer chequeo que se deberá realizar es si el punto dado es interior al área rectangular total de la malla de datos. Es un chequeo de seguridad. Una vez pasado el chequeo se delimita el cuadrado de la malla en el que se encuentra dicho punto. Dicho cuadrado es dividido en dos triángulos por lo que se vuelve a chequear a cuál de las dos áreas triangulares pertenece. Determinada ésta, se obtiene el producto vectorial de los tres vértices que determinará la normal. A partir de la información de los datos de las alturas de los tres vértices se hará una media ponderada para obtener la altitud.

Abierto el entorno y situado el automóvil en la posición inicial, éste es “conducido” mediante un hardware externo: un volante y unos pedales que realizan el juego de acelerador y freno. El cambio de marchas, equivalente al de un vehículo automático, es manejado desde uno de los botones que dispone el volante. Además, al resto de los botones del volante se les ha asociado diferentes cámaras o puntos de vista. Para cada nuevo evento, introducido por el giro del volante, o la introducción de una aceleración o desaceleración, por medio de los pedales, el sistema calcula la nueva posición de acuerdo a estos eventos, la geografía del terreno y, las condiciones del sistema cuerpo.



*Figura 7. “Conducción del automóvil en el terreno”*

## 5. Conclusiones o Consideraciones Finales

Se ha conseguido un avance respecto a trabajos anteriores, al dotar de más capacidades al simulador, consiguiendo actuar sobre terrenos de geografía diferentes, sin necesidad de tener un suelo predefinido para la computación. El proceso de conducción del vehículo se realiza de una manera interactiva mediante un hardware externo fácilmente adquirible en el mercado. La estructura de la programación hace que sea ampliable en el futuro con nuevas funcionalidades.

Como líneas futuras se pretende incorporar hardware específico, como son los cascos de realidad virtual y guantes de datos, que permitan visualizar en un entorno inmersivo el simulador, así como su conducción por el movimiento de las manos. Se pretende cargar entornos en los que se aprecien zonas más urbanizadas, por lo que se introducirá el estudio de colisiones, estado de las superficies sobre las que circula el automóvil, influencia de los estados climatológicos, introducción de avatares, etc.

## Agradecimientos

Este trabajo ha sido posible gracias a la colaboración de Ben Discoe, impulsor del portal de VTerrain.org. Así mismo agradecemos al grupo del Laboratorio de Ingeniería Mecánica del departamento de ingeniería industrial II que ha colaborado en diferentes etapas de este trabajo. Y a la CYCIT que ha subvencionado el proyecto.

## Referencias

- [1] <http://lim.ii.udc.es/research.es.html>
- [2] <http://www.vterrain.org>
- [3] <http://www.usgs.gov/>
- [4] <http://www.wxwindows.org/>
- [5] J. Cuadrado et al., “Modeling and Solution Methods for Efficient Real-Time Simulation of Multibody Dynamics”, *Multibody System Dynamics*, **Vol.** (1) (1997) p. 259.
- [6] J. Cuadrado et al., “Intelligent Simulation of Multibody Dynamics: Space-State and Descriptor Methods in Sequential and Parallel Computing Environments”, *Multibody System Dynamics*, **Vol.** (4) (2000) p. 55.
- [7] BAKKER, E. Y PACEJKA, H.B., (1991) “The magyc formula tyre model”, *Tyre modelling for dynamics analysis, Proceedings 1st International Colloquium on Tyre Models for Vehicle Dynamics Analysis*, Delft, Netherlands, October 21-22, Supplement to Vehicle System Dynamics, Vol. 21, pp. 1- 18.