

# MÁQUINAS DE ESTADO Y VIDEOJUEGOS

Jiménez A., Luque J.

Universidad de Sevilla  
Facultad de Informática  
Dpto. Tecnología Electrónica  
Avda. Reina Mercedes, s/n 41012 SEVILLA

Tfno: (95) 455 27 86

Fax: (95) 455 27 64

## RESUMEN

La enseñanza de los aspectos dinámicos que se describen mediante *Máquinas de Estado*, presenta graves problemas de exposición y aprendizaje. La secuencia temporal de sucesos, la interrelación de los mismos y la conexión con el mundo exterior son conceptos de difícil asimilación, más aún cuando el sistema descrito es de complejidad media o grande. Para solucionar este problema didáctico se ha ensayado, con excelentes resultados docentes, la aplicación de las técnicas de videojuegos al aprendizaje de máquinas de estado.

## 1. INTRODUCCIÓN

En este trabajo se presentan los resultados de una experiencia educativa llevada a cabo en el Departamento de Tecnología Electrónica de la Universidad de Sevilla, para la enseñanza de asignaturas propias del área, a alumnos de la Facultad de Informática.

En concreto, se ha observado que los alumnos encuentran dificultades en el aprendizaje del funcionamiento dinámico de sistemas que habitualmente se describen mediante Máquinas de Estados Finitos (MEF). Entre otras situaciones, las MEF se utilizan principalmente en nuestro área, por una parte para la descripción del funcionamiento de circuitos digitales secuenciales y, por otra, en la caracterización del comportamiento dinámico de protocolos de comunicaciones. Es precisamente en este último aspecto donde, dada la complejidad de las MEF utilizadas, el grado de dificultad para los alumnos resulta más elevado.

Para facilitar el proceso de aprendizaje del funcionamiento dinámico de las MEF se ha desarrollado un proyecto que permite a un instructor la definición de un sistema compuesto por una o varias máquinas de estados. Esta definición incluye, no sólo la descripción textual y procedimental del comportamiento del sistema, sino también una representación gráfica del mismo, así como la definición de la animación que se desea que tenga dicho gráfico en función del comportamiento del referido sistema.

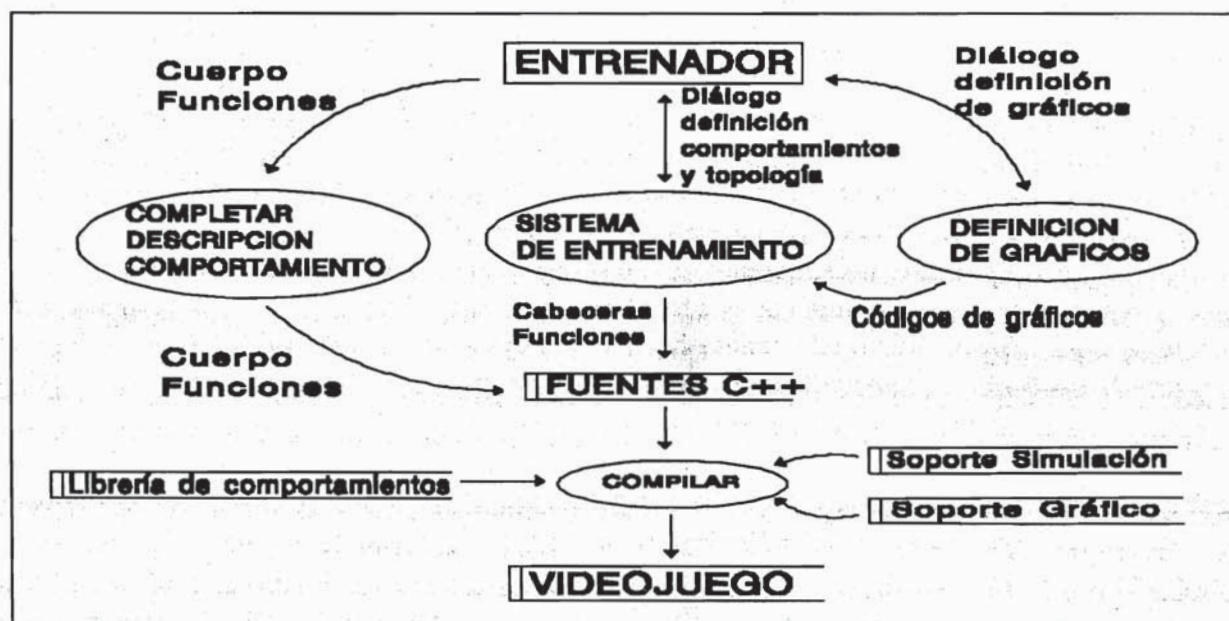
Con todo ello se obtiene un programa que, ejecutado por el alumno, le muestra la evolución dinámica del comportamiento de las MEF. Pero esta presentación no se realiza exclusivamente con gráficos y animaciones convencionales sino que, para darle mayor atractivo, se aprovechan las técnicas de videojuegos. Así, por ejemplo, cuando se describe el comportamiento de un protocolo de comunicaciones, se puede simular que un error afecta a una trama que circula por el canal mediante el "disparo de un proyectil" sobre un "móvil". Éste y otros efectos (sonidos, homúnculos, etc.) dotan a la simulación de una apariencia de videojuego que la hace mucho más atractiva e instructiva para el alumno.

## 2. ANÁLISIS

La gran mayoría de los sistemas de simulación de ámbito general existentes hoy día, están pensados para ofrecer información lo más exacta posible a la realidad, en consecuencia adquieren una complejidad muy elevada tanto en la implementación de nuevos sistemas a simular, como en su manipulación posterior por el usuario final. En estos mismos sistemas, la exposición de los resultados se realiza utilizando largas tablas de datos o gráficas n-dimensionales tales como tensión-tiempo, error-tiempo, etc... Su uso requiere por parte de la persona que los consulta requiere un cierto nivel de conocimientos o experiencia. El método de enseñanza que enfrente al alumno con este tipo de representaciones hacen necesaria en mayor o menor medida, la presencia del profesor.

Estas razones desestimaban la utilización de estos sistemas para nuestros propósitos: obtener aplicaciones de simulación de protocolos de comunicaciones para redes de ordenadores, con fines educativos; determinando la necesidad del desarrollo de un proyecto de sistema generador de aplicaciones de simulación, específico para protocolos de comunicaciones. Los objetivos principales que se establecieron fueron dos y consisten en resolver los dos problemas principales que acusan la mayoría de los sistemas de simulación: *Facilidad* a la hora de especificar los nuevos sistemas que se deseen simular, y *Claridad* en las exposiciones de las simulaciones por las aplicaciones finales obtenidas, suficiente como para que pueda ser empleada en la enseñanza.

La solución adoptada fue utilizar *Máquinas de Estados Finitas* para definir los distintos comportamientos dentro del nuevo sistema y *Técnicas de Videojuegos* para la exposición de los resultados de la simulación. Las MEF que pueden ser integradas dentro del sistema son del tipo extendido, en ellas es posible utilizar variables y establecer prioridades entre las distintas transiciones de un mismo estado. Las técnicas de videojuegos se refieren a un conjunto de implementaciones en lenguaje ensamblador que permiten el movimiento y la animación de un número considerable de gráficos o *sprites*, por toda la pantalla a gran velocidad, con una visualización de gran calidad, sin parpadeo ni desvirtuaciones.



1. DFD del sistema de definición.

El entrenador o instructor es la persona encargada de montar la simulación utilizando las herramientas y servicios que el sistema le proporciona, y obtiene de forma automática un programa ejecutable con la apariencia de videojuego.

En el apartado siguiente se describen brevemente las distintas herramientas con las que el entrenador se deberá enfrentar para el montaje de una nueva simulación.

### 3. DESCRIPCIÓN FUNCIONAL (Instructor)

El instructor o profesor encargado de montar la simulación deberá diseñar el canal de comunicaciones, los comportamientos de cada uno de los nodos de la red, y si lo desea algunos gráficos adicionales para la animación.

Para ello cuenta con tres herramientas de definición, constituidas en tres programas, que son los siguientes:

**1º. Herramienta de definición del Canal de Comunicaciones.**- El entrenador realiza un dibujo de lo que sería la red de ordenadores. Este dibujo puede ser realizado usando cualquier diseñador gráfico que genere una salida en formato PCX.

Posteriormente, a través de esta herramienta, determina dónde se encontrarán, dentro de la pantalla, cada uno de los nodos, y las líneas de conexión entre nodos de la red.

En la definición del canal sólo nos interesa los datos topológicos:

- qué nodo está conectado con qué nodo,
- qué hay conectado a cada extremo de cada línea, (el sistema se encargará de colocar de forma automática los gráficos animados asociados a cada nodo).
- y por último, para cada nodo, el nombre de la MEF que definirá su comportamiento. Para evitar, el tener que definir los comportamientos cada vez que se utilicen, hemos habilitado un mecanismo dentro del sistema para que exista una librería de comportamientos ya definidos, accesibles simplemente con un nombre. Entre ellos se incluyen los que definen los:

- protocolos elementales de enlace: unilateral no restringido, unilateral de parada y espera, unilateral para un canal ruidoso.

- protocolos de ventana deslizante: de un bit, con repetición no selectiva y con repetición selectiva.

- protocolos de acceso al medio con las normas *802.3 Ethernet n-persistente*, *802.4 Token Bus* y *802.5 Token Ring*.

- Incluso existen dentro de la librería comportamientos ideales como *Repetidores*, *Puentes* y *Terminadores*.

Si el instructor desea incluir dentro de esta librería nuevos comportamientos tendrá que hacer uso de la segunda herramienta de definición.

**2º. Herramienta de definición de comportamientos.-** La descripción de los comportamientos se realiza a partir de *Máquinas de Estado*. Dos aspectos caracterizan esta herramienta:

- El primero gráfico, al igual que antes, el instructor debe dibujar por medio de cualquier editor gráfico, el gráfico asociado a la Máquina de Estados. Para después, indicar dónde se encuentran los estados y transiciones. Esta solución nos pareció más sencilla, ya que el problema de generar una salida gráfica de forma automática a partir de las especificaciones de una Máquina de Estados resulta demasiado complejo, de esta forma se deja el diseño del gráfico a cargo del instructor.

- Y el segundo lógico, cada una de las posibles condiciones de entrada y acciones de salida, en una Máquina de Estados, serán funciones escritas en C/C++ por el instructor. Esta herramienta se encargará de generar el código fuente en C++ de todas las cabeceras de dichas funciones. Los cuerpos de estas funciones deberán ser programados, a este efecto el sistema ofrece una serie de servicios para interactuar con la red, que siguen la nomenclatura del modelo de referencia OSI.

Esta estrategia de implementación de máquinas de estado puede resultar al principio limitada al uso de estos servicios, sin embargo la ventaja de esta solución radica en su potencia, estas funciones son generadas con un contenido sin determinar, abierto a todas las posibilidades de un procedimiento en lenguaje C. Con lo cual el ámbito de la definición de comportamientos no queda reducido sólo al mundo de las comunicaciones, sino que da vía libre para ser usado en cualquier otro tipo de entornos.

Es importante hacer notar que cada elemento introducido dentro del sistema por medio de estas herramientas disfrutará durante la simulación de un gráfico animado. La forma y animación de estos gráficos está predefinida por defecto. Sin embargo, si se desean introducir nuevas animaciones es necesario utilizar la tercera herramienta de definición.

**3º. Herramienta de salvaguarda de Gráficos.-** Como ya hemos dicho anteriormente, dentro del programa final obtenido, el sistema ofrece una serie de gráficos, por defecto, con movimiento y animación simultáneos a la simulación. Con esta herramienta el instructor podrá incluir nuevos gráficos. Su diseño se realizará, otra vez más, haciendo uso de la potencialidad de cualquier programa comercial de diseño gráfico, siendo requisito esencial que genere una salida gráfica en el formato estándar PCX. De esta forma, la calidad de los mismos dependerá exclusivamente de las dotes artísticas del que los ejecuta.

#### 4. DESCRIPCIÓN FUNCIONAL (Alumno)

El producto final obtenido por las herramientas de definición consistirá en un programa de simulación de protocolos de comunicaciones dentro de un sistema abierto de intercambio de información, el alumno que lo utilice podrá apreciar fenómenos, tan estudiados como confusos, de colisiones, rebotes, time-out, etc ..., incluso, podrá seguir la evolución de la Máquina de Estado, asociada a cualquier nodo de la red de ordenadores, al mismo tiempo que se está simulando el resto del sistema.

Podemos asegurar que esta aplicación resulta un tanto espectacular. En pantalla se observa cómo "viajan" las tramas de datos, a través de las líneas de la red de ordenadores, la activación de cronómetros, la evolución de las máquinas de estados que se seleccionen, un indicador de la última transición franqueada, la evolución del historial de datos recibidos en cada nodo, además de las animaciones especiales que se hayan introducido.

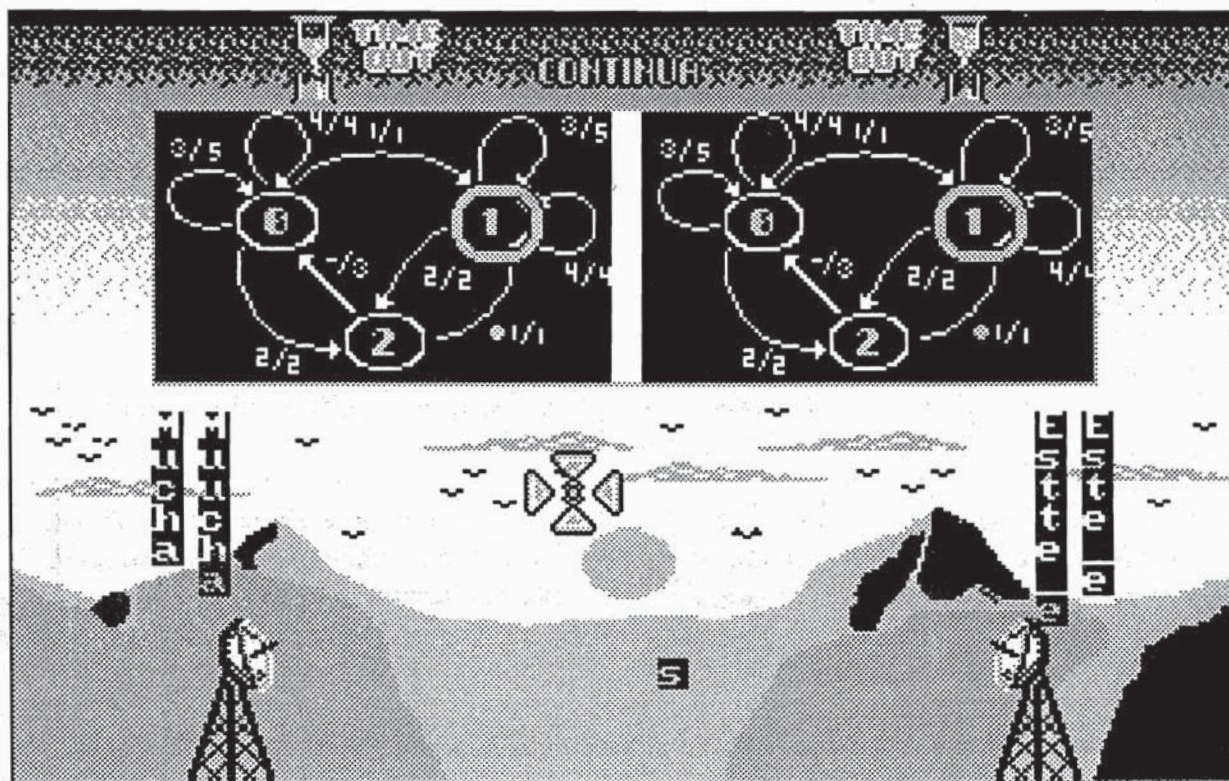
Cada uno de los elementos introducidos tienen asociados uno o más gráficos animados: los ordenadores que se representan como consolas de ordenador o estaciones de microondas, los cronómetros o Time-Outs por medio de relojes de arena, las tramas por caracteres, el estado actual dentro de una máquina de estado por un disco, los terminadores de las líneas por martillos que golpean a las tramas, etc ... Todos se mueven por la pantalla de forma simultánea y totalmente independiente. Una buena y rápida visualización, en la que se pueda distinguir con claridad todos estos elementos dentro de la pantalla, resultaría imposible sin utilizar técnicas de videojuegos.

Hemos querido aprovechar esta circunstancia para potenciar la manera en la que el usuario interactúa con la simulación. Así, el mecanismo para introducir una ráfaga de ruido dentro del canal consiste en lo siguiente: "El alumno dispondrá de un punto de mira que deberá situar, (ayudándose del joystick, el ratón o los cursores), sobre una trama y disparar una vez si desea que ésta se haga errónea, o dos veces para hacerla desaparecer". De esta forma se podrán apreciar los cambios en los comportamientos que definen los protocolos de comunicaciones. Para disminuir la complejidad de manejo del programa, se ha reducido al mínimo la interacción manual por teclado por parte del alumno, librándolo de utilizar largas listas de comandos, hasta conseguir que la única entrada que el alumno tenga que realizar sea la de los mensajes que desee enviar de un nodo a otro de la red.

Pensando en el uso de la aplicación para fines didácticos, existen dos modos de simulación: continua y paso a paso, seleccionables por el alumno en cualquier momento. La primera más rápida, se puede utilizar para acelerar aquellas etapas del protocolo que tenga mejor asimiladas, y la segunda para estudiar con mayor detenimiento las fases del comportamiento todavía confusas por el alumno. El ritmo de la ejecución también es seleccionable entre varios niveles de velocidad.

El alumno cuenta además con un sistema de ayuda asistida, que podrá solicitar en cualquier instante, donde se describirá textualmente, toda la información a cerca de cada una de las transiciones y acciones dentro de la máquina de estados seleccionadas.

A continuación se muestra un ejemplo de programa de simulación obtenido, para un protocolo de ventana deslizante de un bit, con técnica *piggy-backing*.



2. Ejemplo de simulación con técnicas de videojuegos.

En él se observan los siguientes elementos, de abajo a arriba y a ambos lados: los nodos de una red de ordenadores, representados por estaciones de microondas, inmediatamente superior se encuentran los historiales de tramas recibidas del nivel físico y tramas enviadas al nivel de red. En la realidad las tramas consisten en una hilera de bits que pueden contener, a parte de bits de control, muchos caracteres de información alfanumérica o binaria; en la simulación hemos preferido su representación por medio de un único carácter alfanumérico para simplificar su representación. Más arriba, se encuentran los gráficos asociados a las máquinas de estado de cada nodo. En ellos, se encuentra un disco que informa del estado actual de la máquina, (en el ejemplo, las dos máquinas se encuentran en el estado 1), indicadores de última transición activada por medio de un punto. En la parte superior de la pantalla aparecen el indicador de simulación continua y los cronómetros o time-out representados por relojes de arena, obsérvese el detalle de los distintos niveles de arena en cada uno, todo parece indicar que la máquina de la derecha ha sido la última en emitir.

En el programa de simulación las tramas enviadas se mueven a uno y otro lado de la pantalla según sus destinos, éste es el caso de la trama "s" en la parte inferior. En el centro de la pantalla se encuentra el punto de mira que se utiliza para "disparar".

## 5. DESCRIPCIÓN INTERNA

El proyecto se fragmentó en dos subproyectos de complejidad e independencia funcional muy elevada; por un lado el SOPORTE GRÁFICO, y por otro el SOPORTE DE SIMULACIÓN. En lo que respecta a la simulación se consideró conveniente dividir este apartado en dos partes, que se corresponden con los niveles más inferiores del modelo de referencia OSI.

1º. *Nivel Físico* con la Simulación del canal de comunicaciones. En él se tuvieron en cuenta:

- El tráfico de tramas,
- La detección de colisiones,
- Las tramas rebotadas por líneas sin terminadores,
- El marcado y desaparición de tramas erróneas por parte del usuario, y
- El accionamiento de cronómetros o Time-Out.

2º. *Nivel Enlace de Datos* con la Simulación de los comportamientos. Éste supone el nivel más complejo en cuanto a la implementación, se encarga de la emulación de todas las máquinas de estados que se encuentren dentro del sistema, la sincronización de las evaluaciones de las condiciones y las ejecuciones de las acciones de todas las transiciones que se activen.

El proyecto se desarrolla usando la metodología de orientación a objetos. Cada uno de los módulos que lo componen funcionan de forma independiente, gracias a la característica de encapsulación de la definición de clase-objetos. Estos módulos, Gráficos, Red de comunicaciones y Máquinas de estados pueden ser re combinados e integrados dentro de cualquier otro sistema por separado o conjuntamente. Así, es posible:

- Un programa de simulación de protocolos de comunicaciones, usando Gráficos, Red de comunicaciones y Máquinas de Estados.
- Un programa de monitorización, (por ejemplo, una planta industrial), usando Gráficos y Máquinas de Estados.
- Un programa de control, utilizando únicamente los servicios de emulación de Máquinas de Estados.



## 6. CONCLUSIONES

Usando tecnología de videojuegos es posible acaparar con mayor facilidad la atención del alumno, y explicar el comportamiento de un sistema de comunicaciones, o cualquier otro sistema complejo, que de otra forma, supondría un esfuerzo de tiempo considerable por parte del profesor.

Por otro lado, si el profesor elige este método para el aprendizaje de sus alumnos, sólo tendrá que preocuparse del diseño inicial de los comportamientos y del canal de comunicaciones; el programa de simulación generado se encargará de explicar lo que sucede a todos los alumnos de todos los cursos, de todas las promociones, a cualquier hora y en cualquier lugar, en casa o en el propio centro. Nunca una explicación ha tenido tanto rendimiento.

## 7. BIBLIOGRAFÍA

[1] Andrew S. Tanenbaum. "Computer Networks" Prentice-Hall, 1988.

[2] Bradley Dyck Kliwer. "EGA/VGA A Programmer's Reference Guide". Prentice-Hall, 1988.

[3] Peter Coad & Edward Yourdon. "Object Oriented Analysis"