



UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO

FACULTAD DE CIENCIAS

SISTEMA DE MANIPULACIÓN GRÁFICA
DE FENÓMENOS FÍSICOS REALES,
ASISTIDOS POR COMPUTADORA

T E S I S

QUE PARA OBTENER EL TÍTULO DE:

LICENCIADA EN CIENCIAS DE LA COMPUTACIÓN

P R E S E N T A :

MARTHA LORENA EUDAVE LOERA

TUTOR

DR. FERNANDO GAMBOA RODRÍGUEZ

2008



FACULTAD DE CIENCIAS
UNAM



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.



UNIVERSIDAD NACIONAL
AVENIDA DE
MEXICO

FACULTAD DE CIENCIAS
Secretaría General
División de Estudios Profesionales

Votos Aprobatorios

ACT. MAURICIO AGUILAR GONZÁLEZ
Jefe de la División de Estudios Profesionales
Facultad de Ciencias
Presente

Por este medio hacemos de su conocimiento que hemos revisado el trabajo escrito titulado:

Sistema de Manipulación Gráfica de Fenómenos Físicos Reales, Asistidos por Computadora

realizado por **Eudave Loera Martha Lorena** con número de cuenta **0-9528712-2** quien ha decidido titularse mediante la opción de tesis en la licenciatura en **Ciencias de la Computación**. Dicho trabajo cuenta con nuestro voto aprobatorio.

Propietario Mat. María Concepción Ana Luisa Solís González-Cosío

Propietario M. en C. María Guadalupe Elena Ibargüengoitia González

Propietario Tutor Dr. Fernando Gamboa Rodríguez

Suplente M. en I. María de Luz Gasca Soto

Suplente Dr. José de Jesús Galaviz Casas

Atentamente,

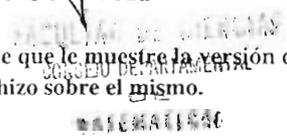
“ POR MI RAZA HABLARÁ EL ESPÍRITU ”

Ciudad Universitaria, D. F., a 18 de septiembre de 2008

EL COORDINADOR DEL COMITÉ ACADÉMICO DE LA LICENCIATURA EN CIENCIAS DE LA COMPUTACIÓN

DRA. ELISA VISO GURÓVICH

Señor sinodal: antes de firmar este documento, solicite al estudiante que le muestre la versión digital de su trabajo y verifique que la misma incluya todas las observaciones y correcciones que usted hizo sobre el mismo.



Sistema de Manipulación Gráfica
de Fenómenos Físicos Reales,
Asistidos por Computadora

Martha Lorena Eudave Loera

Septiembre 2008

*A **Mina**,
por su cariño, comprensión y apoyo incondicional.*

*Al **Señor Eudave** y a **Mi Mami**,[†]
por su fé y por cuidarme siempre.*

Mil Gracias!

Introducción

Durante muchos años, la enseñanza de las ciencias, particularmente de la Física, ha significado un gran reto para los docentes quienes han evidenciado la necesidad de explotar al máximo los recursos didácticos dispuestos a su alcance, entre ellos la computadora que, a través del software educativo, ha jugado un papel muy importante en la presentación de materiales que permiten a los alumnos reforzar los temas revisados en clase y trabajar a su propio ritmo más allá de las instalaciones de un laboratorio escolar.

Desde antes de la inclusión de la computadora al contexto educativo, los profesores han recurrido al uso de diversos materiales didácticos, como murales, maquetas, transparencias, etcétera, para apoyar las actividades docentes a fin de encontrar una manera más efectiva para facilitar el aprendizaje de sus alumnos y cubrir los objetivos que se pretenden alcanzar en cada curso, sin embargo, pese a que ese conjunto de herramientas didácticas son de clara utilidad, no logran por sí mismas promover en el estudiante una participación espontánea ni inmediata, sino que, por el contrario, le otorgan un rol primordialmente pasivo en su proceso de aprendizaje.

Un recurso particularmente exitoso en la enseñanza asistida por computadora en el campo de las ciencias altamente relacionadas con la experimentación, son las simulaciones, especialmente los simuladores interactivos, a través de los cuales se logra estimular el interés y la atención de los estudiantes alentándolos a involucrarse en el proceso de aprendizaje haciendo preguntas o sacando conclusiones, incrementando el rango de análisis de situaciones con las que pueden experimentar. Esto favorece en gran medida la comprensión de fenómenos que, la mayoría de las ocasiones, se manifiestan bajo condiciones que dificultan su observación detallada, ya sea por la imposibilidad de tener acceso real al fenómeno de interés, ya por razones de infraestructura

de laboratorios, ya por limitaciones de orden físico o de seguridad.

En efecto, gracias a su gran potencial gráfico y de exploración, los simuladores interactivos permiten a los alumnos controlar parámetros, manipular objetos y observar las consecuencias de estos cambios en tiempo real, pasando de un medio de representación visual a otro de formalización matemática, estableciendo así una conexión entre el contexto teórico y el práctico en favor tanto de la adquisición como del refuerzo de conocimientos previos a partir de situaciones con las que están familiarizados.

Si bien el uso de recursos tecnológicos en la educación fomenta el desarrollo de software educativo aplicado a actividades específicas, no es posible garantizar que dichas herramientas logren cumplir con los objetivos para los que fueron desarrolladas. Las tecnologías pueden ser eficaces [19] si son concebidas y aplicadas con el propósito de aumentar la inmersión de los alumnos en el aprendizaje, si se ha tomado en cuenta qué es lo que ellos necesitan saber, cómo construyen y comparten sus conocimientos así como la manera en que interactúan utilizando los conocimientos adquiridos.

Aunque existe una gran cantidad de herramientas para implementar estos materiales de apoyo a la educación, desarrollar software educativo de buena calidad y que cumpla con los objetivos para los que fue creado, es una tarea difícil. Uno de los errores más comunes que suelen cometerse durante el desarrollo de este tipo de software es que no se toma en cuenta quién va a usarlo ni en qué contexto, cuáles son sus habilidades y posibilidades. De ahí, surge la importancia de generar productos basados en metodologías de desarrollo que no sólo utilicen la mejor tecnología disponible para el diseño y evaluación de sistemas interactivos, sino que también aseguren la calidad de éstos en términos del apoyo efectivo que brinda al usuario para alcanzar los resultados deseados.

Antecedentes del proyecto

Conscientes de las necesidades que debe cubrir el software educativo, de las ventajas potenciales que ofrece la computadora como herramienta de apoyo a la educación y con la finalidad de minimizar los inconvenientes generados por la falta de un estudio detallado en la elaboración de estas herra-

mientas, en el LIHMM¹ del CCADET², surgió la idea de generar una serie de instrumentos educativos para la enseñanza de la Física que englobaran los aspectos teóricos y prácticos de un curso de Física a nivel medio superior, tomando en cuenta metodologías de desarrollo de software centradas en las necesidades y características del usuario: el *Hipertexto de Física* y el *Laboratorio Basado en Video*. Estos materiales tienen como objetivo permitir a los alumnos estudiar, observar y medir fenómenos físicos con lo que logra cubrirse tanto el aspecto teórico como el práctico del curso.

El *Hipertexto de Física* [25], también llamado *Enciclopedia de la Física*, es una herramienta donde los alumnos podrán consultar de manera sencilla, los tópicos teóricos relacionados con la materia, accediendo a la información de acuerdo a sus necesidades a través de una serie de ligas.

El *Laboratorio Basado en Video (LBV)* [15] es una herramienta que permite el análisis de fenómenos físicos reales tal como estos ocurren en un laboratorio a partir de su observación en video digital. El alumno podrá realizar las mediciones que le ayuden a describir el comportamiento general del sistema que pretende comprender sobre dicho video.

Sin embargo, a pesar de las claras ventajas que representan la *enciclopedia de la Física* y el *LBV* como materiales de apoyo a la educación, era necesario poseer una herramienta que permitiera a los alumnos modificar las condiciones bajo las cuales ocurren los fenómenos físicos presentados en los videos, para que de este modo, ellos mismos pudieran modelar sus propios experimentos, facilitando la comparación entre los resultados obtenidos del estudio de los experimentos reales y los experimentos virtuales.

A partir de dicha necesidad, comienza el desarrollo de un *Sistema de manipulación gráfica de fenómenos físicos reales asistidos por computadora*. Éste es un software educativo multimedia, un simulador interactivo que presenta una serie de experimentos referentes a la asignatura de Mecánica. Los experimentos incluidos en el sistema se desarrollan sobre un ambiente 3D donde sus parámetros pueden adaptarse a los especificados por los alumnos. Además, es posible realizar un mismo experimento bajo diferentes circunstancias e incluso elaborar videos de las simulaciones y visualizarlos en el LVB para su análisis. Así, el sistema de manipulación gráfica es una herramienta

¹LIHMM: Laboratorio de Interacción Humano-Máquina y Multimedia

²CCADET: Centro de Ciencias Aplicadas y Desarrollo Tecnológico, UNAM

muy útil para el refuerzo de conocimientos, y propicia que el alumno tenga un papel activo y autónomo en su proceso de aprendizaje.

Organización de la tesis

Este trabajo se divide en cuatro capítulos: Software educativo, Simulación, Multimedia y ambientes 3D y Desarrollo de un sistema de manipulación gráfica de fenómenos físicos reales, asistidos por computadora.

En el Capítulo 1 “*Software Educativo*”, se analizan tópicos tales como el uso de la tecnología en la educación, el objetivo, funciones, características y tipos de software educativo así como la importancia del uso de metodologías centradas en los alumnos durante la producción de materiales que tienen el objetivo de mejorar y reforzar el proceso de enseñanza-aprendizaje.

En el Capítulo 2 “*Simulación*”, se explica la definición de ésta analizando una serie de modelos que permiten sintetizar sistemas reales; los tipos de simulaciones y el proceso a seguir para desarrollar una simulación. Se analiza la definición de tiempo real en una simulación, las ventajas que representa el uso de los simuladores, sus tipos y características para la observación y el estudio de situaciones complejas. Finalmente, se describen las funciones fundamentales de los motores de simulación física.

En el Capítulo 3 “*Multimedia y Ambientes 3D*”, se analiza la importancia que toman las aplicaciones multimedia y los ambientes tridimensionales para generar aplicaciones de apoyo educativo; el proceso de desarrollo y las herramientas de modelado para la creación de ambientes 3D así como algunas consideraciones sobre los mismos.

Finalmente en el Capítulo 4, “*Desarrollo de un sistema de manipulación gráfica de fenómenos físicos reales, asistidos por computadora*”, se describe detalladamente el proceso de desarrollo del sistema, se retoma también el análisis de las herramientas generadas como antecedente del proyecto. Se define a grandes rasgos el software empleado para su desarrollo. Se estudia la estructura del sistema y las tareas que debe realizar el usuario para trabajar con alguno de los experimentos incluidos en el mismo. Se describen los alcances obtenidos, los trabajos a futuro para mejorar la herramienta y las conclusiones resultantes del desarrollo de este trabajo.

Índice general

Introducción	I
1. Software Educativo	1
1.1. Tecnologías de la educación	2
1.2. La computadora como herramienta educativa	3
1.3. Usos de la computadora en la educación	3
1.3.1. La situación en México	4
1.4. Definición de Software Educativo	7
1.5. Objetivo del software educativo	7
1.6. Funciones del software educativo	8
1.7. Características del Software Educativo	10
1.7.1. Entorno de comunicación o interfaz	10
1.7.2. Bases de datos	11
1.7.3. Motor o algoritmo	12
1.8. Tipos de Software Educativo	13
1.9. Consideraciones para elaborar software educativo	16
1.10. Software educativo centrado en el estudiante	18

1.10.1. Metodología centrada en el estudiante para el desarrollo de software educativo	19
1.11. Comentario Final	21
2. Simulación	23
2.1. Definición de simulación	24
2.2. El uso de simulaciones en el aprendizaje de la física	25
2.3. Sistemas	26
2.3.1. Clasificación de los sistemas de simulación	27
2.4. Modelos	28
2.4.1. Clasificación de los modelos	29
2.5. Clasificación de las simulaciones	31
2.5.1. Simulación orientada a intervalos	33
2.5.2. Simulación orientada a sucesos	35
2.6. Tiempo real	36
2.7. Consideraciones respecto a la simulación	37
2.8. Ciclo de Vida de un proyecto de simulación	40
2.9. Los Simuladores	45
2.10. Tipos de simuladores	47
2.10.1. Diseño	47
2.10.2. Formación	47
2.10.3. Pruebas	48
2.10.4. Mantenimiento	48
2.10.5. Recreación o reconstrucción	49
2.10.6. Ocio y entretenimiento	49

<i>ÍNDICE GENERAL</i>	VII
2.11. Motores de física	50
2.11.1. Funciones de los motores de física	50
2.12. Simuladores de física	51
2.13. Comentario final	51
3. Multimedia y Ambientes 3D	53
3.1. Multimedia	54
3.2. Los sistemas multimedia en la educación	56
3.3. Clasificación de los sistemas multimedia educativos	58
3.3.1. Clasificación según su sistema de navegación	58
3.3.2. Clasificación según su finalidad y base teórica	59
3.3.3. Clasificación según el nivel de control que tiene el desarrollador	61
3.4. Ambientes 3D	62
3.5. Software multimedia para la creación de sistemas 3D	63
3.5.1. Software de modelado	65
3.5.2. Software generador de escenarios sintéticos	66
3.5.3. Software de diseño	66
3.5.4. Software de autoría	67
3.6. Creando ambientes 3D	67
3.7. Comentario Final	68
4. Sistema de Manipulación Gráfica	71
4.1. Antecedentes del proyecto	72
4.1.1. La Enciclopedia de la Física	73

4.1.2. El Laboratorio Basado en Video (LBV)	73
4.2. Comparación con otros simuladores de física	75
4.3. Sistema de Manipulación Gráfica	79
4.4. Características del Simulador	80
4.5. Software utilizado en el desarrollo del sistema	81
4.5.1. Director	82
4.5.2. Shockwave	83
4.5.3. Havok Physics Engine	84
4.6. Estructura del Simulador	84
4.6.1. Creación de la escena	85
4.6.2. Control de objetos y variables físicas	90
4.6.3. Simulación	92
4.6.4. Creación de video	100
4.7. Requerimientos mínimos del sistema	104
4.8. Interacción del usuario con el sistema	105
4.9. Ventajas del Sistema	109
4.10. Comentario final	111
Conclusiones	113
Bibliografía	117

Índice de figuras

1.1. Modelo de un ambiente virtual educativo	17
2.1. Etapas en la explotación de modelos matemáticos	30
2.2. Diagrama de simulación orientada a intervalos	34
2.3. Diagrama de simulación orientada a sucesos	35
4.1. Laboratorio Basado en Video	74
4.2. Relación entre las tres herramientas	81
4.3. Ambiente 3D sobre un espacio Shockwave3D	87
4.4. Tolerancia de colisión	93
4.5. Pasos y subpasos en una simulación	94
4.6. Ciclo de una Simulación	96
4.7. Registro de colisión	98
4.8. Modos de despliegue de simulación	99
4.9. Proceso de control de una simulación	100
4.10. Imágenes almacenadas en la ventana Cast de Director	102
4.11. Imágenes almacenadas en la unidad C del disco duro de la computadora	103

4.12. Constructor de películas AVI	104
4.13. Compresor de video	104
4.14. Diagrama de tareas del usuario	106
4.15. Distribución de controles del simulador	107
4.16. Gráficas de comportamiento de las simulaciones	110

Índice de Tablas

1.1. Proyectos de apoyo tecnológico para la educación en México	5
1.2. Proyectos de apoyo tecnológico para la educación en México II	6
2.1. Etapas de un proyecto de simulación	42
3.1. Etapas de creación de ambientes 3D	69
4.1. Comparación entre Simuladores de Física	78
4.2. Definición y valores de las variables físicas	89
4.3. Movimiento de cámara	90

Capítulo 1

Software Educativo

La introducción del uso de la computadora al sistema educativo se vio favorecida gracias a la disminución de los costos en materia de tecnología, principalmente en hardware, aumentando la posibilidad de que no sólo las instituciones de nivel superior, sino también varias escuelas de nivel preescolar, primaria, secundaria y bachillerato contaran con equipo e instalaciones exclusivas para impartir clases con el apoyo de una computadora.

Como consecuencia, en forma paralela al desarrollo de la informática, se ha incrementado considerablemente el desarrollo de programas computacionales cuyo objetivo principal es auxiliar al profesor en todas aquellas actividades que estén dirigidas al aprendizaje de habilidades específicas y a los alumnos, a reforzar conceptos que ya hayan aprendido. De este modo, el software educativo se ha convertido en una de las principales herramientas de apoyo al proceso de enseñanza-aprendizaje.

Contrario a lo que generalmente se cree, el incremento en el desarrollo de dichos materiales asistidos por computadora no garantiza la efectividad de los mismos, sino que, en muchos casos, pone en evidencia la falta de estrategias que permiten que estas herramientas cumplan con su objetivo; por lo tanto, es de vital importancia desarrollar aplicaciones de calidad que realmente representen un apoyo tanto para los profesores como para los alumnos. Una manera de lograrlo es mediante el análisis de las necesidades que se deben cubrir con el material, de las características de los estudiantes, así como de las posibilidades de desarrollo de software que brindan las herramientas

disponibles.

1.1. Tecnologías de la educación

Tradicionalmente, el papel que el profesor ha desempeñado en la educación es el de transmitir conocimientos a los alumnos sin que necesariamente estos últimos se involucren activamente en el proceso. En la actualidad, dicha concepción del profesor ha cambiado: sigue siendo un transmisor de conocimientos y además su labor consiste en poner al alcance de los alumnos las herramientas necesarias para que puedan reforzar dichos conocimientos de una manera activa.

Un instrumento de gran utilidad al que pueden recurrir los profesores en la búsqueda de opciones más eficaces para lograr la participación de los alumnos es la *tecnología de la educación*. Ésta es considerada como una aplicación sistemática de conocimientos científicos y tecnológicos para la solución de los problemas educacionales.

La expresión *tecnología de la educación* generalmente se utiliza sólo cuando se hace referencia a equipos y máquinas que pueden utilizarse para la enseñanza, como proyectores, pantallas, grabadoras, computadoras o el uso de medios masivos de comunicación como radio y televisión. Sin embargo, la *tecnología de la educación* constituye un instrumento extremadamente útil en la planeación, ejecución y evaluación de sistemas de aprendizaje. Ésta no se reduce a una teoría única, sino que es dinámica y busca incluir innovaciones en los campos de la ciencia y la tecnología, por lo tanto, no determina la adopción o exclusividad de un método o sistema de aprendizaje.

La tecnología [19] aporta nuevos enfoques al aprendizaje, propiciando en muchos casos una mayor interacción con los alumnos. De esta manera, es factible establecer conexiones entre escuelas, maestros y alumnos: se propiciará la participación de los maestros como transmisores de conocimientos y se enfatizará la habilidad para buscar y evaluar información, para explorar preguntas abiertas, para resolver problemas, para desarrollar el pensamiento crítico, para formar y construir conocimientos nuevos a fin de alcanzar un nivel nuevo de comprensión. La metodología a emplear en el uso de estas tecnologías dependerá de los objetivos a alcanzar, de las características de

las poblaciones y de las variables incluidas en el proceso. Esto indica que, mientras el profesor comprenda mejor el significado de la tecnología de la educación, tendrá más libertad para poder crear y desarrollar aquellos sistemas de aprendizaje que mejor se ajusten a sus necesidades.

1.2. La computadora como herramienta educativa

Sierra [43] comenta que las primeras concepciones del uso de la computadora en la educación estaban apoyadas en un modelo de enseñanza donde se veía al profesor como transmisor de conocimientos y al alumno como un receptor generalmente pasivo.

Esta situación afortunadamente ha cambiado. El papel de la computadora ha de definirse dentro de la metodología actual de la enseñanza: no puede ni debe sustituir al profesor en el desempeño de su función docente. En la metodología de enseñanza-aprendizaje más reciente, el profesor es aquel que propicia las condiciones necesarias para que el alumno se responsabilice de su propio aprendizaje.

1.3. Usos de la computadora en la educación

La computadora como herramienta de apoyo en la educación, no se limita al uso de los programas educativos instalados en ella, también comprende su uso en los laboratorios no sólo como un instrumento capaz de organizar diferentes medios de comunicación – audio, video e Internet – sino también como un medidor de tiempo, tensión eléctrica, intensidad de corriente, intensidad luminosa, nivel de pH, sonido, temperatura, etcétera.

Gracias a su capacidad de analizar datos, la computadora permite contrastar la teoría con el trabajo experimental y facilita el análisis de los fenómenos a partir de las ideas previas de los alumnos. El trabajo en el laboratorio puede ser mucho más eficiente ya que el tiempo dedicado a la toma de datos y análisis matemático puede automatizarse, permitiendo, de

este modo, que los profesores dediquen más tiempo a la interpretación de los experimentos realizados.

La importancia de los laboratorios escolares en la enseñanza de ciencias experimentales en cualquier nivel educativo es indiscutible. El trabajo práctico en el laboratorio proporciona al alumno la experimentación y el descubrimiento personal: evita el concepto de “resultado correcto” que se tiene cuando éste aprende sólo los datos de un libro y no tiene la oportunidad de aprender directamente a partir de experimentos. No obstante, el uso de los laboratorios requiere precauciones adicionales a las de una clase convencional para que los alumnos descubran por sí mismos y aprendan de sus propios errores.

La manera en que el estudiante descubre el mundo y lo que es relevante para una investigación particular, depende de la experiencia previa que él tenga. Si el alumno no conoce el concepto que espera ver, fracasa la interpretación de un experimento. Muchas veces este conocimiento se consigue con una mezcla de demostración y discusión, pero hay además, otras razones importantes que justifican el trabajo empírico: el desarrollo de habilidades prácticas, de capacidades para resolver problemas científicos y sensibilidad para apreciar el trabajo de los científicos.

1.3.1. La situación en México

El uso de recursos tecnológicos como apoyo a la enseñanza, ha evolucionado de manera importante en México. En un principio se establecieron los “Laboratorios de Computación” en escuelas privadas donde en la mayoría de los casos, el profesor fue ignorado puesto que generalmente se contrataba a ingenieros en computación para impartir las clases de computación; posteriormente, como un segundo intento por introducir el uso de la computadora en los centros de educación básica, media y media superior, numerosas instituciones trabajaron en conjunto para desarrollar programas educativos donde se contempla la participación activa tanto de alumnos como de profesores. En las Tablas 1.1 y 1.2 se presenta una cronología de dichos proyectos.

Actualmente, el uso de la computadora en la escuela se ha convertido en un asunto de gran importancia no sólo por la cantidad de computadoras instaladas en escuelas públicas y privadas, sino también porque se ha logrado

AÑO	INSTITUCIÓN	PROYECTO
1978	Academia de Investigación Científica UNAM ¹ , SEP ² , IPN ³ , Fundación Arturo Rosen- blueth	Programa televisivo “Domingos en la Ciencia” Grupos de investigación dedicados al estudio de la interacción de los niños con la computadora.
1984	UNAM y la Academia de Investigación Científica	“Primer Simposio Internacional: La Computación en la Educación Infantil”.
1985 - 1990	SEP y el ILCE ⁴	Desarrollo de un modelo pedagógico y programas educativos. COEEBA ⁵ - SEP: Instalación de computadoras en secundarias como apoyo didáctico en el salón de clases y para la enseñanza de LOGO y BASIC. Desarrollo de programas educativos en las áreas de Español, Matemáticas, Ciencias Sociales y Ciencias Naturales.
1994	SEP UNAM	Introducción de “Educación Tecnológica en Computación” a nivel secundaria. Incorporación de la materia de Informática en la Escuela Nacional Preparatoria.

Tabla 1.1: Proyectos de apoyo tecnológico para la educación en México

¹ Universidad Nacional Autónoma de México (UNAM)

² Secretaría de Educación Pública (SEP)

³ Instituto Politécnico Nacional (IPN)

⁴ Instituto Latinoamericano de Comunicación Educativa (ILCE)

AÑO	INSTITUCIÓN	PROYECTO
1996	Gobierno Federal	RedEscolar ⁶ : Educación a distancia mediante programas de televisión (EduSat) y el uso de computadoras conectadas en red.
1998, a la fecha	Gobierno Federal, SEP, ILCE, IPN, UPN ⁷ , UNAM, entre otros	Enciclomedia ⁸ [40]: sistema que integra y articula medios, recursos y herramientas relacionados con la educación primaria; se fundamenta en los libros de texto gratuitos y que, a partir de su edición digital, los enlaza a la biblioteca de aula, a fotografías, mapas, visitas virtuales, videos, audio, interactivos y otros recursos; es un espacio de colaboración organizada entre profesores, instituciones públicas, privadas e individuos. Aprovecha e integra recursos y experiencias de otros proyectos importantes como RedEscolar, Sepiensa ⁹ , BibliotecaDigital, SEC21, Enseñanza de la Física con Tecnología y Enseñanza de las Matemáticas con Tecnología ¹⁰ , entre otros.

Tabla 1.2: Proyectos de apoyo tecnológico para la educación en México II

atender tanto a la instrucción en temas propios de la tecnología informática, así como el uso de la computadora como auxiliar didáctico mediante el uso

⁵ Computación Electrónica para la Educación Básica

⁶ <http://redescolar.ilce.edu.mx/>

⁷ Universidad Pedagógica Nacional (UPN)

⁸ <http://www.enciclomedia.edu.mx/>

⁹ <http://www.sepiensa.org.mx/>

¹⁰ <http://www.efit-emat.dgme.sep.gob.mx/>

de software educativo como enciclopedias interactivas, programas para hacer mapas conceptuales, programas para desarrollar habilidades en pensamiento estratégico y programas multimedia interactivos para la enseñanza de física y matemáticas.

1.4. Definición de Software Educativo

El *software educativo* según Morales [28], es un recurso tecnológico constituido por programas computacionales de aplicación didáctica, es decir, aquellos programas que han sido diseñados para la presentación, el desarrollo de actividades y el apoyo a los procesos educativos. Estos programas computacionales son utilizados tanto por el profesor para enseñar como por los alumnos para aprender, para alcanzar determinados propósitos y principalmente para reforzar algunos conocimientos vistos en clase.

Además, Galvis [12] sugiere que el término *software educativo* se aplica tanto a los programas que apoyan la administración de procesos educativos o de investigación como a los que dan soporte al proceso de enseñanza-aprendizaje.

Es necesario aclarar que de acuerdo con Sierra [43], todos los programas computacionales de aplicación didáctica, exceptuando los lenguajes de programación y excluyendo también las aplicaciones como los procesadores de textos, las bases de datos, las hojas de cálculo, etcétera, son designados generalmente como programas educativos.

1.5. Objetivo del software educativo

El software educativo tiene como objetivo auxiliar al profesor en las actividades dirigidas a la enseñanza de habilidades específicas y ayudar a los estudiantes con su aprendizaje, proporcionándoles los materiales, ejemplos, ejercicios, tareas, etcétera, que les permita construir, organizar y refinar su propio conocimiento.

Para trabajar de una manera adecuada, el software educativo no sólo debe

ser funcional, sino también atractivo y estar al alcance de los estudiantes, en el sentido que pueda captar su atención y guiarlos en la resolución de las tareas propuestas. Además, debe permitir que los estudiantes cometan errores para que, al analizarlos, logren una mejor comprensión y sean encaminados hacia un verdadero entendimiento de los conceptos expuestos.

Gamboa [15] señala que algunas ventajas del software educativo respecto a otros medios de enseñanza consisten en la posibilidad de ser estructurado de forma tal que el punto de partida pueda escogerse libremente; la información debe presentarse de manera variada tanto tiempo y tantas veces como sea necesario, de modo que se facilite la asimilación de los conocimientos que se pretende transmitir; finalmente, el software puede incluir cuestionarios dirigidos y simulaciones interactivas que permitan complementar este esquema.

En el desarrollo de software educativo no sólo es importante la preparación multimedia del material a incluir, sino su estructura, la elección de medios y las estrategias pedagógicas seleccionadas. A un usuario puede serle completamente inútil un software educativo si éste no presenta los temas bien estructurados, o si los ejercicios no resaltan adecuadamente los aspectos fundamentales para su aprendizaje. Así, un sistema bien diseñado debe permitir al usuario interrumpir una lección en cualquier momento y continuar posteriormente; de esta manera, el usuario tiene la posibilidad de avanzar a su propio paso, manipular variables, observar sus efectos y relacionarlos con su realidad, etcétera.

1.6. Funciones del software educativo

Marquès [26] indica que, de acuerdo a la forma de uso que le de cada profesor, el software educativo cumple las siguientes funciones:

Informativa: Proporciona información estructurada de la realidad que se pretende representar. Algunos ejemplos son las bases de datos, los simuladores y los tutoriales.

Instructiva: Orienta y regula el aprendizaje, promueve la participación de los estudiantes con la finalidad de facilitar el logro de objetivos educativos específicos; además, puede disponer un tratamiento global de

la información. Los tutoriales son los que realizan de manera más explícita esta función instructiva porque dirigen las actividades de los estudiantes en función de sus respuestas y progresos.

Motivadora: Incluye elementos para captar la atención de los alumnos y enfocarlos hacia los aspectos más importantes de cada actividad a desarrollar, siendo esta función extremadamente útil para los profesores.

Evaluada: La interactividad propia de estos materiales permite responder inmediatamente a las acciones de los estudiantes, haciéndolos adecuados para evaluar su trabajo. Esta evaluación puede ser de dos tipos: implícita, cuando el estudiante detecta sus errores y se evalúa a partir de las respuestas que le da el sistema o explícita, cuando el programa presenta informes valorando la actuación del alumno.

Investigadora: Promueve la investigación para que los estudiantes profundicen más en algún tema propuesto en el software, ya sea buscando determinada información o bien, cambiando las variables de un sistema.

Expresiva: Los estudiantes se expresan y se comunican con la computadora y con otros compañeros a través de las actividades propuestas en los programas, además, las computadoras no permiten ambigüedad expresiva en sus diálogos con los estudiantes, de manera que los alumnos se ven obligados a cuidar más la precisión de sus mensajes.

Lúdica: Trabajar con las computadoras realizando actividades educativas, es una labor que a menudo tiene unas connotaciones lúdicas para los estudiantes, quienes sienten mayor motivación al aprender jugando.

Innovadora: Aunque no siempre sus planteamientos pedagógicos resulten innovadores, los programas educativos se pueden considerar materiales didácticos con esta función, ya que utilizan la tecnología más reciente incorporada a los centros educativos y, en general, suele permitir muy diversas formas de uso. Esta versatilidad propicia amplias posibilidades de experimentación didáctica e innovación educativa en el salón de clases.

1.7. Características del Software Educativo

El software educativo, señala Marquès [27], puede tratar las materias de matemáticas, idiomas, física, geografía, dibujo, etcétera, de maneras muy diversas, ya sea a través de cuestionarios que faciliten información estructurada a los alumnos o mediante la simulación de fenómenos, entre otros y ofrecer un entorno de trabajo más o menos sensible a las circunstancias de los alumnos, también puede ser rico en posibilidades de interacción; pero cumple en general con cinco características esenciales:

1. Está comprendido por un conjunto de materiales desarrollados con una finalidad didáctica.
2. Utiliza la computadora como el soporte en el que los alumnos realizan las actividades propuestas.
3. Es interactivo porque responde a las acciones del estudiante y permite un diálogo o intercambio de información computadora-estudiante.
4. Individualiza el trabajo del estudiante ya que se adapta al ritmo de trabajo de cada alumno.
5. Es fácil de usar porque los conocimientos de computación necesarios para manejarlo son mínimos.

La mayoría de los programas didácticos, al igual que muchos de los programas en computadora desarrollados sin alguna finalidad educativa, tienen tres módulos bien definidos: el módulo que gestiona la *comunicación con el usuario* mediante los dispositivos de entrada y salida; el módulo de los contenidos informativos del programa, como las *bases de datos*; y el módulo que gestiona las respuestas del sistema a las acciones de los usuarios, es decir, el *motor del sistema*.

1.7.1. Entorno de comunicación o interfaz

La interfaz es el entorno a través del cual los programas establecen el diálogo con sus usuarios y es la que posibilita la interactividad característica de estos materiales. Está integrada por dos sistemas:

1. El sistema de comunicación *programa-usuario*, que facilita la transmisión de la información entre la computadora y el estudiante a través de pantallas, altavoces, sintetizadores de voz, módems, convertidores digitales-analógicos, entre otros.
2. El sistema de comunicación *usuario-programa*, que facilita la transmisión de información del usuario hacia la computadora mediante el uso del teclado y el ratón, micrófonos, pantallas sensibles al tacto, lápices ópticos, módems, convertidores analógico-digitales, etcétera.

Con la ayuda de técnicas de inteligencia artificial y el desarrollo de las tecnologías multimedia, se investiga la elaboración de entornos de comunicación cada vez más intuitivos y capaces de proporcionar un diálogo abierto y próximo al lenguaje natural.

1.7.2. Bases de datos

Contienen la información específica que cada programa presentará a los alumnos; proporcionan datos organizados en un entorno estático para facilitar su exploración y hacer consultas selectivas. Pueden estar constituidas por los siguientes datos:

Modelos de comportamiento: representan la dinámica de los sistemas, entre ellos se distinguen a los modelos físico-matemáticos y a los modelos no deterministas, los primeros tienen leyes perfectamente determinadas por medio de ecuaciones, los segundos son representados por ecuaciones con variables aleatorias, por gráficos y por tablas de comportamiento.

Datos de tipo texto: representan la información alfanumérica.

Datos gráficos: pueden ser dibujos, fotografías, secuencias de video, objetos 3D, entre otros.

Sonido: secuencias de audio.

Las bases de datos pueden tener una estructura *jerárquica*, si existen elementos subordinados de los que dependen otros subordinados, como organigramas; *relacional*, si están organizados mediante fichas o registros con una misma estructura y rango o bien puede tener una estructura *documental*, si utiliza descriptores cuya finalidad es almacenar grandes volúmenes de información documental como revistas o periódicos.

Según la forma de acceder a la información se pueden distinguir dos tipos:

Bases de datos convencionales. Tienen la información almacenada en carpetas, mapas o gráficos que el usuario puede recorrer según su criterio para recopilar información.

Bases de datos tipo sistema experto. Son bases de datos muy especializadas que recopilan toda la información existente de un tema concreto y además asesoran al usuario cuando accede buscando determinadas respuestas.

1.7.3. Motor o algoritmo

El motor del programa gestiona, en función de las acciones de los usuarios, las secuencias en que se presenta la información de las bases de datos y las actividades que pueden realizar los alumnos.

Se distinguen cuatro tipos de algoritmos o motores:

Lineal: Presentan al alumno una secuencia de información y ejercicios dependiendo de sus respuestas; su interactividad es pobre y el programa se hace tedioso.

Ramificado: Siguen recorridos pedagógicos diferentes según el juicio que hace la computadora sobre las respuestas de los alumnos o dependiendo de la decisión del alumno de profundizar más en ciertos temas; ofrecen mayor interacción y suelen estructurar los contenidos en niveles de dificultad.

Tipo entorno: proporcionan herramientas de búsqueda y procesamiento de la información que los alumnos pueden utilizar libremente para construir las respuestas a las preguntas del programa, es decir, el estudiante elige qué ha de hacer y cuándo lo ha de hacer. Este es el caso de los entornos de resolución de problemas *problem solving* donde los estudiantes conocen parcialmente la información necesaria para su resolución y han de buscar la información que falta, aplicar reglas, leyes y operaciones para encontrar la solución. Existen cuatro tipos de entornos:

Estático: si el usuario únicamente puede consultar y en algunos casos aumentar o disminuir la información que proporciona el entorno, pero no puede modificar su estructura.

Dinámico: si el usuario, además de consultar la información, también puede modificar el estado de los elementos que configuran el entorno.

Programable: si a partir de una serie de elementos el usuario puede construir diversos entornos.

Instrumental: si ofrece a los usuarios diversos instrumentos para realizar determinados trabajos.

Tipo sistema experto: también llamados *intelligent tutoring systems*, están elaborados en base a técnicas de inteligencia artificial. Tienden a reproducir un diálogo entre el programa y el estudiante pues pretenden comportarse como lo haría un tutor humano. Guían a los alumnos paso a paso en su proceso de aprendizaje, analizan su estilo de aprender así como sus errores; proporcionan en cada caso la explicación o ejercicio más conveniente.

1.8. Tipos de Software Educativo

Los programas educativos a pesar de poseer algunos rasgos esenciales y una estructura general común, tienen características muy diversas: unos aparentan ser un laboratorio, una enciclopedia o una biblioteca, otros se limitan a ofrecer una función instrumental del tipo máquina de escribir o calculadora, otros se presentan como un juego, un libro, un aparente examen, un simulador de fenómenos y situaciones o como una combinación de

características, pero sin importar su apariencia, cada uno de ellos obedece a objetivos específicos.

A continuación se describirán brevemente los tipos de software educativo que de acuerdo con Sierra [43], los más conocidos son de ejercicios y prácticas, programas de presentación-demostración, tutoriales, programas de simulación, sistemas expertos, sistemas de diálogo y juegos interactivos también llamados programas lúdicos.

Ejercicios y prácticas por computadora: El propósito principal de estas aplicaciones es auxiliar al profesor en las actividades dirigidas al aprendizaje de habilidades específicas. Los ejercicios sencillos pueden resultar muy eficaces como forma de trabajo y deben adaptarse flexiblemente a la secuencia y metodología de cada profesor. Propician el aprendizaje por descubrimiento donde las prácticas se pueden extender aún más allá del uso de una computadora y desarrollarse por medios convencionales como lápiz y papel.

Programas de presentación-demostración: La mayoría de estos programas suelen limitarse a presentar nueva información, ilustrar conceptos previamente estudiados mediante gráficos y animaciones o bien, presentan actividades, menús de opciones, etcétera. Permiten un mínimo de interacción y son aptos para ser utilizados directamente por el profesor frente a la clase a modo de pizarrón electrónico. Las posibilidades que ofrece la computadora de incluir representaciones gráficas, con animación, movimiento y efectos sonoros, junto con su gran rapidez y capacidad de cálculo, son características a tener en cuenta al evaluar el uso de lenguajes y sistemas para crear este tipo de programas.

Programas tutoriales: Son programas que en mayor o menor medida dirigen el trabajo de los alumnos. Pretenden que, a partir de cierta información y mediante la realización de ciertas actividades previstas de antemano, los estudiantes pongan en juego determinadas capacidades; aprendan o refuercen conocimientos. Comparan las respuestas de los alumnos con los patrones que tienen como correctos, guían el aprendizaje de los estudiantes; facilitan la realización de prácticas más o menos rutinarias y su evaluación.

De acuerdo al manejo de errores que cometen los estudiantes, estos programas pueden ser directivos o no directivos:

Programas directivos: hacen preguntas a los estudiantes y controlan en todo momento sus actividades. La computadora adopta el papel de juez que examina al alumno. Se producen errores cuando la respuesta del alumno está en desacuerdo con la que la computadora tiene como correcta. La mayoría de estas aplicaciones fracasa.

Programas no directivos: la computadora adopta el papel de un laboratorio o instrumento a disposición de la iniciativa de un alumno que pregunta y tiene una libertad únicamente limitada por las normas del programa. La computadora no juzga las acciones del alumno, se limita a procesar los datos que éste introduce y a mostrar las consecuencias de sus acciones sobre el entorno. Objetivamente no se producen errores, sólo desacuerdos entre los efectos esperados por el alumno y los efectos reales de sus acciones sobre el entorno. No está implícita la noción de fracaso. El error es sencillamente una hipótesis de trabajo que no se ha verificado y que se debe sustituir por otra. En general, siguen un modelo pedagógico, potencian el aprendizaje a través de la exploración, favorecen la reflexión, el pensamiento crítico y propician el uso del método científico.

Programas de simulación: Con esta estrategia [6] la computadora es utilizada para modelar el mundo real o algún conjunto de condiciones reales de manera que éstas puedan ser manipuladas y estudiadas de una manera más sencilla. El alumno puede probar sus hipótesis cambiando las variables del modelo y observar los efectos causados en el ambiente de simulación; de éste modo la computadora se convierte en un micro laboratorio. Estos programas exploran situaciones presentadas mediante secuencias gráficas fijas o dinámicas. El objetivo de esta modalidad de aprendizaje es reflejar la importancia de los distintos factores que intervienen en un determinado proceso, y descubrir la naturaleza de las condiciones que posibilitan su modificación.

Sistemas expertos: Son programas basados en una forma de programación que tiene la capacidad de aprender nuevos datos o relaciones durante su ejecución. Su desarrollo se halla estrechamente relacionado con el campo de la investigación en inteligencia artificial (IA). Por ejemplo, un sistema experto para enseñar álgebra puede detectar que el alumno

llega al resultado correcto pero no reduce las expresiones a su forma más simple, entonces el programa establece una relación con lecciones sobre reducción de expresiones o hace sugerencias al estudiante para que éste descubra cómo puede reducir sus resultados.

Sistemas de diálogo: Estos sistemas tienen dos modalidades: cuando la computadora pregunta y cuando el que pregunta es el estudiante. Cuando la computadora tiene la iniciativa, el programa presenta un cuestionario seleccionado al azar, entre las muchas preguntas que el sistema es capaz de formular a partir de unos datos básicos. En la segunda modalidad, el estudiante decide interrogar al programa sobre cualquier información y el sistema puede contestar a preguntas cuyas respuestas no hayan sido previstas, mediante inteligencia artificial.

Juegos educativos: El elemento lúdico de estos sistemas suele convertir un ejercicio en un desafío motivador en donde el alumno considera a la computadora como un adversario al que puede ganar. Una de las desventajas que se presentan en los juegos educativos es que tanto el alumno como el profesor pueden confiar demasiado en el funcionamiento automático de las actividades como mecanismo de aprendizaje. Para evitar el riesgo de un aprendizaje poco profundo, es conveniente que las clases incluyan actividades adicionales para consolidar lo aprendido por medio de la computadora.

1.9. Consideraciones para elaborar software educativo

El uso de las computadoras en la educación ha evolucionado sobre dos líneas: una que enfatiza los aspectos teóricos del conocimiento, generalmente a través de tutoriales o enciclopedias y la otra que se enfoca en la experimentación mediante simulaciones. Tanto los tutoriales como los materiales de referencia en pocas ocasiones incluyen un componente experimental y la mayoría de las simulaciones no ofrecen una explicación teórica. El resultado es que los estudiantes aprenden teoría que difícilmente pueden transferir a situaciones reales relacionadas con su experiencia cotidiana, o tienen experiencias que no pueden explicar o generalizar.

1.9. CONSIDERACIONES PARA ELABORAR SOFTWARE EDUCATIVO 17

La forma en que un software educativo puede compensar esta limitación es a través de la creación de un “ambiente de aprendizaje”, es decir, una combinación entre la explicación de un problema y la experimentación con el mismo.

Cuando se elabora software educativo, se debe asegurar que éste será utilizado para la tarea apropiada. En la Figura 1.1 se representa el modelo de un ambiente virtual educativo; en éste se identifica el ciclo de un sistema donde el usuario introduce los parámetros de entrada que modificarán el estado inicial del ambiente virtual; estos datos serán procesados en la simulación¹¹ y posteriormente, el resultado de tales cambios se deberá reflejar en pantalla; el usuario, en este caso el estudiante, deberá tener la oportunidad de repetir el proceso cuantas veces crea necesario.

Con el fin de proporcionar una perspectiva educativa, es importante controlar la simulación mediante el apoyo de un *programa tutorial* que esté apegado al programa de estudios y cuyo propósito sea conducir al estudiante en todo momento a través de la aplicación.

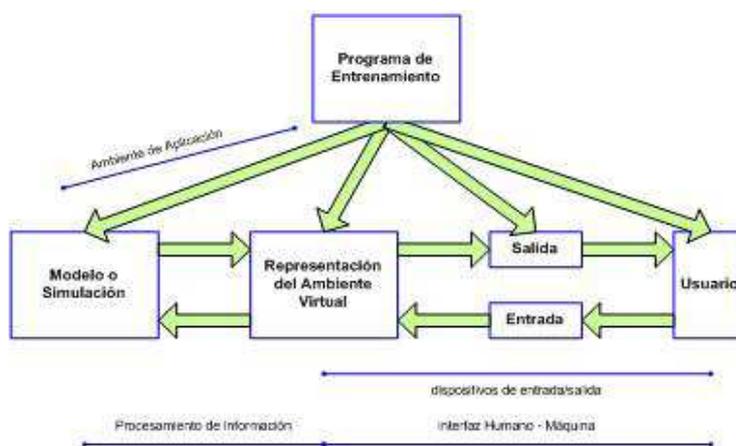


Figura 1.1: Modelo de un ambiente virtual educativo

Es necesario tener en cuenta el comportamiento dinámico en respuesta a las entradas dadas por el usuario; se pueden elaborar modelos matemáticos muy complejos, pero lo importante es la manera en que están asociados con una representación principalmente visual y si es necesario auditiva, de los

¹¹Simulación: Representación matemática del sistema que será analizado.

objetivos didácticos del sistema. Para los estudiantes puede ser más sencillo entender los temas a partir de diagramas o modelos que a través de salidas numéricas. En este sentido, la simulación es una herramienta que facilita la exploración de situaciones que pueden ser complejas o no, permitiendo al usuario equivocarse y aprender de sus errores, con lo cual mejorará su nivel de comprensión y también entenderá mejor la operación del mismo.

Para introducir los parámetros que definirán el comportamiento de la simulación, el usuario se vale de dispositivos de *entrada/salida* los cuales, no sólo son el teclado, el ratón y el monitor, sino también, dependiendo de la aplicación, pueden usarse cascos y guantes. Sin embargo, la tecnología utilizada en cada aplicación es de suma importancia; puede significar la diferencia entre un sistema útil o inútil: por lo tanto, es importante asegurarse de que tales tecnologías sean compatibles con las necesidades y posibilidades del usuario.

Más allá de la tecnología utilizada, una interfaz inapropiada puede inhibir el proceso de aprendizaje e incluso perder por completo el objetivo del sistema, que en este caso es auxiliar al alumno en la comprensión de un problema, de ahí la importancia de tomar en cuenta que los sistemas no sólo deben cumplir con dicho objetivo, más allá de eso, deben ser herramientas que los alumnos puedan comprender y utilizar.

1.10. Software educativo centrado en el estudiante

La introducción de nuevas tecnologías y el desarrollo de software educativo como herramientas de apoyo, ha significado un gran cambio en la manera de presentar la información que los alumnos deben entender y manejar, sin embargo, el software educativo ha mostrado un desempeño que no está a la altura de las expectativas que generó: el uso de la computadora no incrementa el porcentaje de aprendizaje de los estudiantes.

Gamboa [13] comenta que este problema de bajo desempeño se debe a que el desarrollo de software educativo está basado sobre una falsa premisa: “las computadoras son fáciles de usar, así, el software educativo *debe ser fácil de utilizar*”. Esta apreciación surge a partir de la creencia de que las

1.10. SOFTWARE EDUCATIVO CENTRADO EN EL ESTUDIANTE 19

nuevas tecnologías resolverán de facto todos los problemas de usabilidad de las computadoras; pero lo cierto es que cada nueva tecnología trae consigo problemas que necesitan especial atención para evitar el desarrollo de sistemas que resulten difíciles de entender e inútiles para comunicar conceptos al usuario.

Durante el desarrollo de software educativo, generalmente suele prestarse mayor atención al aspecto gráfico e interactivo de las aplicaciones, dejando de lado el análisis de aquellas herramientas que sean las más apropiadas para la solución de problemas específicos de acuerdo a los objetivos que se deban cumplir, y además, tampoco se toman en cuenta las posibles dificultades a las que se pueda enfrentar el alumno para entender cómo funciona el programa, ocasionando que éste no ponga atención en la comprensión del material que contiene.

Por lo tanto es necesario reflexionar acerca de la estructura del sistema, la forma de evaluarlo, analizar la manera en la que trabajan los estudiantes e involucrarlos desde las primeras fases del desarrollo del software, de tal manera que los programadores y diseñadores puedan orientarse en el planteamiento de objetivos para implementar prototipos que sean refinados hasta lograr un balance entre las necesidades, posibilidades y restricciones del sistema con el desempeño del alumno.

1.10.1. Metodología centrada en el estudiante para el desarrollo de software educativo

Entre las consideraciones que deben estudiarse en el desarrollo de software educativo, es la de incluir ejercicios con una fuerte referencia a la vida diaria del alumno porque así le será más fácil asimilar las tareas asignadas y entenderá la utilidad de resolverlas. Además, se le deberá proporcionar acceso a la ayuda en todo momento y se le mostrará evidencia para que pueda autoevaluar si resolvió satisfactoriamente el problema pero sin incluir mensajes de felicitación o reprimenda.

Para obtener un sistema de calidad, Gamboa [13] asegura que se debe poner especial atención a la *interfaz del sistema*, la cual, mediante un buen diseño logrará el objetivo del software. Para ello, es necesario que el equipo de

desarrollo responda las siguientes preguntas: ¿qué es lo que se quiere enseñar? ¿en qué contexto? ¿a quién se le quiere enseñar? y ¿cómo se le quiere enseñar? con ello se construyen objetivos o metas de usabilidad de software educativo centrado en el usuario.

Cualquier instrumento educativo debe ser planeado, diseñado, desarrollado y evaluado de acuerdo a una metodología centrada en el usuario que describe cómo se toman las diferentes decisiones para obtener software educativo que no sólo tome ventaja de las tecnologías de interacción sino que garantice que el producto final sea de calidad, con ello se logrará definir *cómo se quiere enseñar*.

Para definir *a quién se le va a enseñar*, el sistema debe estar acorde a la realidad, a las necesidades del país y desarrollado para un público en particular.

Por otro lado, se debe poner especial atención en la estructura didáctica del material y ésta deberá ser revisada por especialistas, con esto se define *qué es lo que se quiere enseñar*.

Finalmente, *el contexto en que se debe enseñar* dependerá de la situación educativa que se deba apoyar, es decir, si el sistema será utilizado dentro de un curso formal sometido a la coordinación y evaluación del profesor, si el alumno lo utilizará por interés personal o por curiosidad.

Ahora bien, para responder adecuadamente todas las preguntas planteadas, se deberá asegurar que el sistema cumpla con las siguientes características:

- **Util:** Debe integrar las funciones y funcionalidades necesarias en la tarea del usuario, es decir, el sistema hace lo que debe hacer y sirve para lo que está hecho.
- **Usable:** Debe estar organizado y diseñado de tal manera que el usuario sea capaz de entender su estructura y trabajar con él.
- **Promover un conocimiento profundo:** El software educativo debe ser una guía que no sólo repita el contenido de un libro, sino que ofrezca a los estudiantes los elementos (ejercicios, experimentos, metodologías, editores) que promuevan la reflexión acerca de un fenómeno particular y su abstracción.

Siguiendo los puntos anteriores, es posible generar herramientas educativas más eficaces en apoyo a la enseñanza porque se ha tomado en cuenta qué es lo que el alumno necesita saber y cómo debe manejarlo, poniendo a su alcance únicamente aquella información que le es útil al igual que las funcionalidades incluidas en los sistemas desarrollados, todo esto en un contexto adecuado.

1.11. Comentario Final

En éste capítulo se ha hecho notar que en la actualidad, la computadora juega un papel de suma importancia en la educación, ya sea para su aprovechamiento como un instrumento de laboratorio que permita contrastar la teoría con el trabajo experimental o como un medio para presentar alguna aplicación de apoyo al profesor sin llegar a sustituirlo.

Como consecuencia, la cantidad de aplicaciones desarrolladas con fines educativos se ha incrementado considerablemente, convirtiendo el uso de software educativo en una herramienta utilizada con mayor frecuencia porque es un recurso tecnológico que proporciona los materiales, ejemplos y ejercicios que le permiten al alumno intervenir directamente en su proceso de aprendizaje, presenta la información en forma variada y cuantas veces sea necesario, estableciendo de esta manera, un puente natural entre la forma tradicional de educación y el uso de las nuevas tecnologías.

A pesar de todos los esfuerzos realizados en la difusión de programas de calidad, es necesario poner más atención en el proceso de elaboración de software educativo, analizando detenidamente qué se quiere enseñar, quién va a aprender, la manera en que se puede adaptar el material a las necesidades y prioridades del usuario final, que en este caso es el alumno, sin distraerlo ni alejarlo de los objetivos que se quieran alcanzar, desarrollando productos basados en metodologías de diseño, desarrollo y evaluación que, además de utilizar la tecnología disponible, aseguren la calidad del mismo.

Capítulo 2

Simulación

La simulación es una de las herramientas fundamentales empleadas en la toma de decisiones apoyadas por computadora. Hace posible responder a preguntas del tipo: ¿qué pasa si...? o ¿qué se debe hacer para...? Promoviendo de esta manera, especialmente en el alumno, el aprendizaje significativo por descubrimiento.

A lo largo del tiempo, el concepto de simulación ha cambiado. Puede considerarse que desde mediados del siglo XX, durante el desarrollo de las computadoras, la simulación se encontraba en una etapa de gestación, pues su uso más extendido estaba asociado principalmente a representaciones físicas, modelos a escala, tácticas y estrategias militares, donde su aplicación desde el punto de vista matemático, era casi nula.

Posteriormente, la simulación se utilizó como una técnica de resolución de modelos matemáticos para los que no existían procedimientos analíticos válidos, o por lo menos eficientes, enfatizando los procesos algorítmicos que permitían la optimización de tales modelos; también ha sido utilizada con fines descriptivos para conocer el funcionamiento de un sistema como punto de partida para mejorar la salida del modelo asociado.

Actualmente, la simulación es una de las herramientas de análisis cuantitativo más utilizadas en campos como la Economía, la Estadística, la Física, la Informática y muchas otras áreas altamente ligadas a la experimentación. Históricamente se ha visto como una tecnología autosuficiente [20], donde el

usuario puede ser la única persona interactuando con el sistema.

Por otra parte, la simulación tiene enormes aplicaciones industriales, económicas, educativas, comerciales, etcétera. Es un instrumento de entrenamiento en el manejo de dispositivos, sustancias, maquinaria, sistemas de control, e incluso de procedimientos quirúrgicos; es una herramienta de predicción bursátil, económica, química y meteorológica, como apoyo educativo y terapéutico además de ser una herramienta de entretenimiento entre muchas otras aplicaciones. Por lo tanto, la simulación toma un sentido crítico en la evaluación del comportamiento de diferentes escenarios involucrados en la resolución de un problema.

2.1. Definición de simulación

Rodríguez [39] indica que según West Churchman, el término simulación se define de la siguiente manera: “X simula a Y si y sólo si:

1. X y Y son sistemas formales
2. Y se considera como el sistema real
3. X se toma como una aproximación del sistema real
4. Las reglas de validez en X no están exentas de error”

En ésta definición se representa a la realidad como un sistema y la creación de un sistema análogo o modelo que debe contener las características esenciales de la realidad pero con un menor grado de complejidad, sin embargo, no se toma en cuenta el concepto de tiempo, que es una variable altamente significativa cuya importancia sí es ponderada por otros autores.

Tal es el caso de Ríos [38], Banks [3], Harrel [20] y Naylor [29], para quienes la simulación es una metodología que consiste básicamente en la construcción de un modelo computacional que no sólo debe imitar sino también describir la parte esencial del comportamiento de un sistema dinámico o parte de él, *a lo largo del tiempo* con la finalidad de manejar experimentos. A través de la manipulación de los parámetros de dicho modelo o simulador, el

usuario podrá experimentar y observar la reacción del sistema dadas ciertas entradas, extraer conclusiones de acuerdo a los resultados obtenidos que deberán ser analizados estadísticamente para estimar la evolución del sistema real en intervalos determinados y con ello inferir el comportamiento general del fenómeno o proceso en estudio.

2.2. El uso de simulaciones en el aprendizaje de la física

Anteriormente se ha mencionado la posibilidad de utilizar simulaciones como reemplazo o como auxiliar de las experiencias de laboratorio, lo cual brinda enormes posibilidades dado el carácter irrestricto de la variación de parámetros que se permiten. Sin embargo, de acuerdo con Kofman [24], para evitar la imagen de artificialidad que se puede generar en el alumno, se aconseja vincular, en la medida que sea materialmente posible, la simulación con la experiencia.

Mediante la comparación de la experiencia real con la simulación, el alumno logra tener una idea más clara del significado del modelo matemático a través de la observación de cómo éste se aproxima a la realidad, pero que bajo ciertas condiciones puede tener importantes discrepancias, es por ello que resulta útil comparar distintos modelos matemáticos con un sólo modelo físico, para verificar el rango de validez de cada uno.

Arruda [2] apunta que para entender el problema de la enseñanza de la Física, es necesario buscar la causa y el efecto de la situación del proceso de enseñanza-aprendizaje de esta ciencia, donde se ven involucrados aspectos epistemológicos, psicológicos y didácticos. Estos factores parecen reforzar la idea de que la Física impartida tanto a nivel medio superior como superior, no parece ejercer influencia significativa en la formación cultural científico-técnica del estudiante, tampoco está contribuyendo satisfactoriamente al aprendizaje de conceptos y leyes ni impulsa a los alumnos a razonar la solución de problemas de Física o de la vida diaria.

Ahora bien, dado que la Física es una ciencia experimental, los experimentos de laboratorio con manipulación de los sistemas, observación directa de los fenómenos y mediciones de diversas magnitudes, resultan de fundamental im-

portancia para la enseñanza, sobretodo en su ciclo básico. En caso contrario, se corre el riesgo de adquirir una noción distorsionada o *matematizada* de la propia disciplina, perdiéndose de vista que los modelos matemáticos, como una representación parcial e incompleta de la complejidad del mundo físico, están siempre subordinados a los fenómenos reales. Por ello, de acuerdo con Zamarro [45], es importante explotar las tecnologías que pueden contribuir a la enseñanza de la Física, como por ejemplo, mediante la simulación de fenómenos físicos, en el caso en que estas simulaciones permitan interactividad y diversas representaciones de los datos, porque pueden convertirse en eficaces laboratorios virtuales en los que el alumno puede investigar, modificar parámetros, realizar medidas, representar el fenómeno de estudio y sus datos de formas muy diversas.

Considerando que posiblemente los estudiantes pueden pensar que la computadora automáticamente resolverá los problemas propuestos, es preciso guiarlos a la reflexión de encuadrar su resolución en la teoría física para obtener expresiones matemáticas, en donde la computadora sí proporcionará una alternativa a la resolución de dichas expresiones así como la representación del fenómeno y de sus resultados.

Una vez obtenida la simulación, es fundamental que el alumno analice los resultados y profundice la investigación del fenómeno, aprovechando las capacidades de modificación de parámetros del modelo. De esta manera puede comprender mejor el fenómeno en estudio dado que además va a obtener una representación dinámica en tres dimensiones que evidencian de modo inmediato la repercusión que la manipulación de parámetros tiene sobre el sistema.

2.3. Sistemas

La realidad se describe como un sistema que consiste en una colección de *objetos o entidades* que interactúan para alcanzar un cierto objetivo, entonces, un sistema es la parte de la realidad o el proceso sobre el que se muestra interés [38].

La colección de objetos de un sistema puede ser tan sólo un subconjunto de los objetos que deberían ser considerados en un sistema real, es decir,

deben constituir un sistema simplificado. Cada una de esas entidades puede cambiar a lo largo del tiempo; a dicho cambio se le denomina *evento o suceso*, que es una ocurrencia instantánea que altera el estado del sistema cuya evaluación dependerá del valor de sus atributos, es decir, el *estado de un sistema*, que es el conjunto mínimo de variables (de estado) necesarias para caracterizar o describir todos aquellos aspectos de interés del sistema en un cierto instante.

Por otra parte, García [17] indica que hay tres módulos básicos para adquirir información de un sistema con el objetivo de conocer su comportamiento o modificarlo: experimentar, analizar y simular.

1. **Experimentar:** consiste en estudiar el sistema real en funcionamiento y tomar medidas directamente. Es el método más preciso, pero no siempre es posible llevarlo a cabo dado que hay experimentos que son demasiado peligrosos, costosos o simplemente no es posible disponer del sistema real.
2. **Analizar:** se refiere al uso de modelos basados en el razonamiento para calcular directamente los tiempos del sistema con la finalidad de simplificar el sistema real, donde es posible asumir cosas que no siempre se corresponden con la realidad, en consecuencia, cualquier variación del modelo resulta complicada de realizar pues además consume mayor cantidad de tiempo de desarrollo.
3. **Simular:** es el proceso donde se trata de imitar el comportamiento dinámico de un sistema con el fin de llegar a conclusiones aplicables al mundo real a través del análisis de la variación de sus parámetros.

2.3.1. Clasificación de los sistemas de simulación

Atendiendo a la relación entre la evolución de las propiedades de interés y la variable independiente tiempo, Guasch [18] plantea una clasificación de los sistemas en *continuos, discretos, orientados a eventos discretos y combinados*:

- (a) **Sistemas continuos:** el sistema en estudio se considera como un flujo continuo de información, es decir, las variables de estado del sistema

evolucionan a lo largo del tiempo. Un ejemplo de este tipo de sistemas es el cambio de temperatura en una habitación durante cualquier intervalo o bien la variación del nivel de líquido en un tanque.

- (b) **Sistemas discretos:** se caracterizan porque las propiedades de interés del sistema cambian únicamente en un cierto instante o en determinada secuencia de instantes, y permanecen constantes el resto del tiempo, siguiendo por lo general un patrón periódico.
- (c) **Sistemas orientados a eventos discretos:** son un caso particular de los sistemas discretos. Se caracterizan porque las propiedades de interés del sistema cambian únicamente en una secuencia de instantes y se puede considerar que permanecen constantes el resto del tiempo pero obedeciendo a un patrón aleatorio.

Ejemplos de estos sistemas son las líneas de producción, las redes de computadoras, los sistemas de control de tráfico, entre otros. Un proceso muy conocido que ilustra el comportamiento de un sistema orientado a eventos discretos es el peaje de una autopista, al cual llegan coches de modo aleatorio y en el cual se desea prever el tiempo de espera de los coches en las colas en función del número de taquillas abiertas. Considerando los cambios de estado en una taquilla (libre, ocupado), puede observarse que los instantes en los cuales aparecerá un cambio obedecen a un patrón aleatorio.

- (d) **Sistemas combinados o híbridos:** son aquellos que combinan subsistemas que siguen filosofías continuas o discretas respectivamente. Es el caso de los sistemas que poseen componentes que deben ser necesariamente modelados según alguno de dichos enfoques específicos.

2.4. Modelos

El acceso a la observación y al estudio directo o detallado de un sistema real es muy complicado debido a las condiciones en que se desarrolla; más difícil aún es modificarlo para observar su comportamiento; por esta razón, un sistema se estudia a través de un modelo.

El *modelo de un sistema* es una representación abstracta y simplificada del sistema real, de un proceso o de una teoría. Debe describir (modelar) parcial o totalmente, las características de interés así como los mecanismos internos del sistema para prever sus respuestas y controlarlo bajo diversas situaciones, lo que propicia una mejor comprensión de su funcionamiento.

Un modelo puede ser la construcción a escala de un prototipo, una maqueta, una representación analógica de un sistema mediante circuitos eléctricos, una representación gráfica bidimensional o tridimensional o bien puede ser una descripción cualitativa del sistema. Al proceso de desarrollo de estas representaciones se le llama *modelado* [18].

2.4.1. Clasificación de los modelos

Existen varios tipos de modelos utilizados para representar los sistemas: físicos, mentales, simbólicos, entre otros. Cuando los modelos son utilizados en entornos de simulación digital, es necesario que éstos formalicen el conocimiento que se tiene del sistema en modo conciso y sin ambigüedades, es decir, que tengan una interpretación única además de ser procesados por una computadora. Estas características determinan el uso de modelos simbólicos matemáticos como herramienta para representar las dinámicas de interés de cualquier procedimiento en un entorno de simulación digital; mapean las relaciones existentes entre las propiedades físicas del sistema que se pretende modelar en las correspondientes estructuras matemáticas cuya formalización dependerá de las características intrínsecas de las dinámicas que se quieran representar [18].

La descripción de un problema real en términos matemáticos, no es una metodología de trabajo propia de la simulación digital, sino que es inherente a la mayoría de las técnicas que se utilizan para resolver cualquier tipo de problema, las cuales suelen seguir unas pautas que, de modo general, se resumen en el diagrama de la Figura 2.1.

Aunque existe una gran diversidad de metodologías para el desarrollo de modelos matemáticos de sistemas físicos, también debe considerarse lo siguiente para garantizar una representación eficiente del sistema real:

- Un modelo se desarrolla siempre a partir de una serie de aproxima-

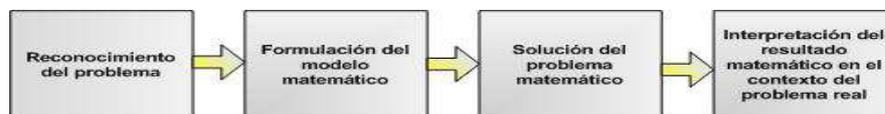


Figura 2.1: Etapas en la explotación de modelos matemáticos

ciones e hipótesis que, consecuentemente, representan parcialmente la realidad.

- Un modelo se construye para una finalidad específica y debe ser formulado para que sea útil a dicho fin.
- Un modelo debe ser simple y recoger todos los aspectos esenciales del sistema en estudio representándolos adecuadamente.
- Debe ser una representación abstracta de la realidad lo suficientemente sencilla como para facilitar su mantenimiento, adaptación y reutilización.

Atendiendo a las características que debe poseer un buen modelo, así como los objetivos del estudio de simulación, los modelos de simulación suelen clasificarse de diferentes formas: estáticos, dinámicos, deterministas, estocásticos, continuos y discretos.

Modelos estáticos respecto a modelos dinámicos: Los *modelos estáticos* suelen utilizarse para representar el sistema en un cierto instante; en su formulación no se considera el avance del tiempo. Este tipo de modelos es muy útil cuando el sistema se encuentra en equilibrio (no evoluciona respecto al tiempo). Si se cambia el punto de equilibrio alterando uno o más de los valores del sistema, el modelo permite deducir el resto de los valores, pero no muestra la manera en que cambiaron.

En contraposición a los modelos estáticos, los *modelos dinámicos* permiten deducir cómo las variables de interés del sistema en estudio evolucionan respecto al tiempo.

Modelos deterministas respecto a modelos estocásticos: Un modelo se denomina *determinista* si su nuevo estado puede ser completamente definido a partir del estado previo y de sus entradas, es decir, ofrece

un único conjunto de valores de salida para un conjunto de entradas conocidas.

Los *modelos estocásticos* requieren de una o más variables aleatorias para formalizar las dinámicas de interés. En consecuencia, el modelo no genera un único conjunto de salida cuando es utilizado para realizar un experimento, sino que los resultados generados sirven para estimar el comportamiento real del sistema.

Modelos continuos respecto a modelos discretos: Los *modelos continuos* se caracterizan por representar la evolución de las variables de interés de forma continua. En general suelen utilizarse ecuaciones diferenciales ordinarias si se considera simplemente la evolución de una propiedad respecto al tiempo, o bien ecuaciones en derivadas parciales si se considera también la evolución respecto al espacio.

De modo análogo a la definición de modelos continuos, los *modelos discretos* se caracterizan por representar la evolución de las variables en forma discreta en intervalos determinados.

A partir de la clasificación de modelos, es posible describir un sistema continuo mediante un modelo discreto y viceversa. La decisión de utilizar alguno de ellos, depende de los objetivos particulares de cada estudio y no tanto de las características del sistema.

2.5. Clasificación de las simulaciones

Varios autores especializados plantean un gran número de clasificaciones relevantes de las simulaciones, sin embargo, estas carecen de dimensiones bien definidas. De Jong [8] comenta que algunas veces las características técnicas de las simulaciones juegan un papel importante en la clasificación, otras veces es la manera en que el estudiante puede interactuar con una simulación, por lo tanto, plantea una clasificación en la que se distinguen cuatro tipos de simulaciones: *simulaciones físicas, de procedimientos, de situaciones y de procesos*.

Además de la clasificación sugerida por DeJong [8], algunos autores como Ríos [38], definen también una clasificación de las simulaciones de acuerdo al

tipo de modelo que se haya elegido para representar el sistema y dan especial importancia a la *simulación de eventos discretos* y a la *simulación continua*.

- **Simulaciones físicas:** Son simulaciones que enseñan acerca de algo. Los objetos o fenómenos físicos se representan gráficamente sobre ambientes en dos o tres dimensiones y la interacción con el usuario no es imprescindible. En el aspecto gráfico pueden ser tan realistas como se requiera, sin embargo, es importante que el sistema se comporte del mismo modo en que lo haría en el mundo real.
- **Simulaciones de procedimientos:** Su propósito principal es que el usuario aprenda una secuencia de acciones que constituyen un procedimiento. Ejemplos comunes de este tipo de simulación pueden ser enseñar a utilizar una calculadora o un teléfono, diagnosticar el mal funcionamiento de un equipo o cómo aterrizar un transbordador espacial, entre otros.
- **Simulaciones de situaciones:** Son aquellas que representan las actitudes y el comportamiento que deben asumir las personas ante una situación o problemática. En todas las simulaciones de situaciones, el usuario es el integrante más importante, ya que es éste quien toma el rol principal, mientras que los otros roles son manejados por estudiantes que interactúan en el mismo programa o por la computadora.
- **Simulaciones de procesos:** Son simulaciones que se utilizan generalmente para informar al estudiante acerca de un proceso o concepto que no se manifiesta visiblemente, por ejemplo, los simuladores de procesos sirven para pronosticar el sistema económico de un país, cómo la ley de la oferta y demanda afecta el precio de un producto o cómo crece o disminuye la tasa poblacional de una región.

Además de la clasificación sugerida por DeJong , algunos autores como Ríos , definen también una clasificación de las simulaciones de acuerdo al tipo de modelo que se haya elegido para el sistema y dan especial importancia a la *simulación de eventos discretos* y a la *simulación continua*.

- **Simulaciones de eventos discretos:** La evolución en el tiempo de un sistema se traduce en cambios en algún atributo de alguna entidad

que se denomina suceso y que ocurre en algún instante. El tiempo entre dos instantes se denomina *intervalo*. El estado de una entidad en un intervalo se denomina *actividad*. Si los sucesos que acotan la actividad causan cambios, el estado de la entidad se denomina *actividad del objeto*. El conjunto de operaciones que transforman el estado de una entidad se denominan *acciones*. Un proceso describe la sucesión de estados por los que pasa una entidad durante su vida en el sistema.

La simulación de eventos discretos emplea métodos estadísticos para generar comportamientos aleatorios y estimar el desempeño de los modelos empleados [20].

Puesto que la evolución en el tiempo en un modelo implica estructuras lógicas complejas de sus elementos, es necesario recurrir a estrategias para describir esta evolución; los sucesos que la definen, se generan en distintos instantes y el paso del tiempo se controla mediante un *mecanismo de reloj o reloj de la simulación*. Esencialmente, hay dos tipos de mecanismo de reloj que identifican dos estrategias de simulación que son: la *simulación orientada a intervalos* y la *simulación orientada a sucesos* [38].

2.5.1. Simulación orientada a intervalos

En el método de simulación orientada a intervalos, el tiempo de simulación avanza según pasos fijos Δt , es decir, de t a $t + \Delta t$, a $t + 2\Delta t$, ...

El diagrama de la Figura 2.2 muestra una simulación con incremento de tiempo uniforme. Una vez iniciado el reloj, se considera una condición de finalización, si esta se da (V), se imprimen los resultados y termina la simulación; si no se cumple (F), el reloj avanza y guarda en memoria las estadísticas. A continuación, el sistema cambia de estado si se produce algún suceso y se comprueba de nuevo la condición de término.

Obviamente, en la simulación orientada a intervalos únicamente se detectan los sucesos ocurridos en el intervalo $(t, t + \Delta t)$ en el instante $t + \Delta t$, con lo que se introducen errores en la simulación. Otro inconveniente en el método es que si los tiempos entre sucesos son muy grandes en comparación con Δt , se invierte esfuerzo computacional al actualizar el reloj de simulación sin que se produzcan cambios en el sistema.

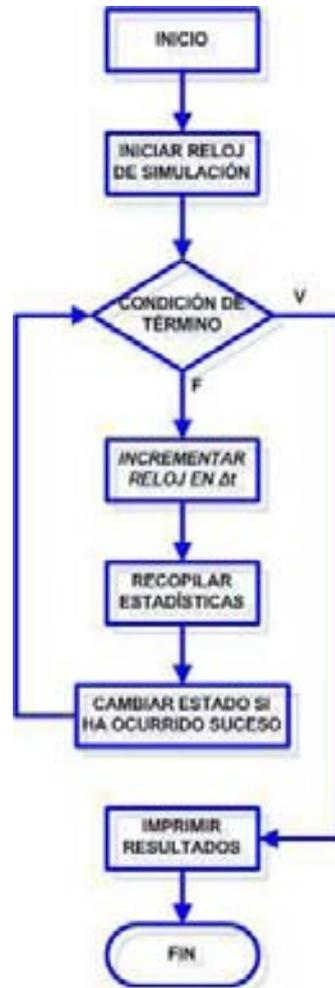


Figura 2.2: Diagrama de simulación orientada a intervalos

Por tal motivo, a pesar de su sencilla implementación y de que se trata de una estrategia valiosa en la simulación de sistemas continuos, se prefiere para la simulación de eventos discretos la segunda alternativa: el método de simulación orientada a sucesos.

2.5.2. Simulación orientada a sucesos

En el método de simulación orientada a sucesos, el tiempo de simulación avanza del instante actual t al instante t' del siguiente suceso; como puede observarse en la Figura 2.3, a diferencia del método de simulación síncrona, el cambio de estado se obtiene cuando se produce un suceso.



Figura 2.3: Diagrama de simulación orientada a sucesos

En éste método de simulación sólo se representan explícitamente los sucesos en el modelo. Los periodos entre cada uno de los sucesos son insignifican-

tes, pues se puede considerar que no consumen tiempo de cómputo, aunque las actividades entre sucesos consuman tiempo en el mundo real.

- **Simulación combinada discreta-continua:** En este tipo de simulación, algunas variables de estado cambian continuamente con respecto al tiempo y otras cambian instantáneamente en momentos separados. En estos sistemas las variables de estado continuas y discretas interactúan generalmente de tres maneras:
 - Un evento discreto puede causar un cambio discreto en el valor de una variable de estado continua.
 - Un evento discreto puede causar que la relación que gobierna una variable de estado continua cambien en un instante en particular.
 - Una variable de estado continua puede provocar que un evento discreto ocurra.

2.6. Tiempo real

Cuando se dice que una simulación funciona en tiempo real, esta consiste en mostrar imágenes del estado del sistema a medida que éstas se van produciendo, ya que ésta es su característica principal, de manera que sirva para que reflejen visualmente los cambios producidos por las acciones del usuario. Estos cambios por lo general suelen responder a la representación de un fenómeno real cuya evolución se intenta replicar, proporcionando la sensación de continuidad.

Con el término *tiempo real*, se asume la existencia de las siguientes restricciones sobre el comportamiento del sistema:

1. El **tiempo de respuesta** es el periodo que transcurre entre la entrada de un dato y la obtención de una salida; en las simulaciones en tiempo real este intervalo se debe reducir al máximo, lo cual ayudará a que el observador se sienta inmerso en el sistema.
2. La **frecuencia de salida** es un parámetro totalmente diferente al retraso y depende también del intervalo de respuesta entre dos salidas que

son consecutivas del sistema. En una representación gráfica, la frecuencia de salida constituye la **frecuencia de actualización de imagen**. En la mayoría de las simulaciones se debe especificar la restricción de que haya una frecuencia de refresco mínima e incluso constante.

3. La correspondencia entre el **tiempo aparente** de la simulación y el **tiempo físico** para el observador son otras restricciones temporales independientes, ya que los objetos deben moverse en la simulación con la misma velocidad que lo harían en el mundo real, además de que se deben **sincronizar** las diferentes salidas que corresponden al mismo fenómeno, como por ejemplo el sonido y los gráficos.

Se llaman *restricciones* porque son valores que deben tomarse en cuenta para hacer los ajustes necesarios en la simulación, y hacer que esta logre dar la apariencia de estar reaccionando en tiempo real a las acciones introducidas por el usuario.

2.7. Consideraciones respecto a la simulación

Una simulación por sí misma no es una herramienta de solución de problemas [20], más bien es una herramienta de evaluación y apoyo al observador porque imita lo que sucede en un sistema real, o lo que es percibido por el sistema que está en la fase de diseño. Los datos de salida de la simulación deben corresponder directamente a las salidas que se pudieron registrar del sistema real. Pedgen [33] señala que es posible desarrollar un modelo de simulación sin suposiciones ambiguas o modelos matemáticamente solubles, pero también existen desventajas muy fuertes.

Presnell [37] indica que existen tres aspectos principales que han contribuido al incremento del uso de las simulaciones en diferentes disciplinas:

1. El incremento de la velocidad de las computadoras y la reducción en los costos de hardware.
2. La adición de salidas gráficas y de animaciones permite llevar a la simulación a salas de sesiones en donde los resultados se presentan por medio

de instrumentos de animación en la pantalla de la computadora, en lugar de presentarse como informes en papel; de esta forma, la audiencia puede absorber la información en forma más rápida y clara.

3. El desarrollo de herramientas de simulación fáciles de usar que permiten analizar resultados estadísticos completos.

Las técnicas de simulación por computadora se utilizan satisfactoriamente por las siguientes razones citadas por Pedgen [33], Harrell [20], Banks [3] entre otros autores:

- La simulación es lo suficientemente versátil para modelar cualquier sistema.
- Muestra el comportamiento del sistema a lo largo del tiempo.
- La observación de los cambios que se obtienen de un simulador a partir de experimentar con ciertos parámetros del modelo, así como los cambios en las entradas, permiten inferir aspectos relativos a la sensibilidad del sistema y qué variables son las que más pueden beneficiar el rendimiento del mismo.
- Es menos costosa y consume menos tiempo que experimentar con el sistema real.
- La simulación es visualmente más atractiva y atrapa el interés del usuario.
- Proporciona resultados que son más sencillos de entender y comunicar.
- La simulación suele ser utilizada con una perspectiva pedagógica para ilustrar y facilitar la comprensión de los resultados que se obtienen mediante técnicas analíticas.
- Es factible explorar procedimientos de operación, reglas de decisión, flujos de información y procedimientos organizacionales sin romper la continuidad de las operaciones del sistema real.
- Se puede probar el diseño de nuevo hardware, esquemas físicos y sistemas de transporte sin comprometer recursos de adquisición.

- Es posible probar la veracidad de las hipótesis referentes a cómo o por qué ocurren ciertos fenómenos.
- El tiempo puede comprimirse o expandirse permitiendo la aceleración o el retraso de un fenómeno bajo investigación.
- Intuir la manera en la que interactúan las variables del sistema.
- Intuir la importancia de las variables en el desempeño del sistema.
- El estudio de una simulación puede ayudar a entender cómo opera el sistema.
- Es posible responder a la pregunta ¿qué pasa si...? lo cual es muy útil en el diseño de nuevos sistemas.
- La simulación permite estudiar o experimentar con sistemas cuyo grado de complejidad es muy variado.
- Es posible o deseable utilizar la simulación para comprobar o verificar soluciones analíticas.
- En instituciones de enseñanza, la simulación es un valioso auxiliar pedagógico que permite que los estudiantes ejerciten y desarrollen habilidades básicas de análisis y toma de decisiones.
- La simulación permite visualizar en los sistemas los efectos inducidos por alteraciones estructurales o funcionales sin afectar al sistema real.
- La aplicación de ésta técnica y el análisis cuidadoso de los resultados permite normar criterios para la toma de decisiones, así como instrumentar acciones específicas permitiendo evaluarlas en periodos cortos o largos.
- Es un auxiliar para el proceso de planeación de cualquier organización [39].

Es importante a notar que, de acuerdo con Guasch [18], la popularización de las técnicas de simulación ha sido paralela al incremento de potencia y flexibilidad de las nuevas herramientas que salen al mercado. Su facilidad de manejo así como sus entornos de visualización/animación han facilitado

la incorporación de nuevos usuarios menos “especializados”. Si bien se ha popularizado su utilización, *también se ha incrementado su mal uso*. Un modelo de simulación no puede ser mejor que los datos y las técnicas empleadas en su construcción. Además, trabajar con simulaciones trae consigo varios problemas que, de no ser tratados cuidadosamente y a tiempo, puede traer como consecuencia no solo que no se logren los objetivos que se pretendía alcanzar sino también, pérdida de tiempo y dinero. Se pueden citar entonces, las siguientes desventajas:

- Generalmente para su uso se requieren más recursos económicos y de tiempo.
- No existe una certeza absoluta sobre los resultados obtenidos.
- Es necesario dar un “mantenimiento” continuo al modelo para que no se vuelva obsoleto ante un medio ambiente muy dinámico o poco estable [39].

Por otro lado, una de las mayores desventajas de las simulaciones, comenta Pedgen [33], consiste en que por lo general no se tiene una idea clara de cómo funciona el sistema que se quiere simular, esto es, el no saber qué variables afectan el sistema y cuáles de ellas son más significativas, provocando que los eventos a simular no queden bien delimitados y se produzcan errores demasiado costosos.

Una vez analizadas las ventajas y desventajas del uso de simulaciones, se amplía la posibilidad de crear materiales de apoyo a la educación de mejor calidad, explotando sus alcances para que estos sean aprovechados en el cumplimiento de los objetivos a alcanzar.

2.8. Ciclo de Vida de un proyecto de simulación

Guasch [18] señala que un proyecto de simulación es dinámico por naturaleza. Los resultados que se van obteniendo a medida que el proyecto se va desarrollando, ponen de manifiesto nuevos problemas, así como limitaciones

inherentes al sistema en estudio, y pueden forzar a reconsiderar la orientación tomada en un principio.

La Tabla 2.1 muestra el conjunto de etapas de un proyecto de simulación. Aunque puede parecer que el desarrollo de un proceso de simulación es secuencial, en la práctica no es así. Por ejemplo, si el modelo de simulación obtenido no supera la etapa de validación (Etapa 5), es posible que sea necesario modificar tanto el modelo conceptual como el de simulación.

- 1. Formulación del problema:** en la fase inicial de cualquier proyecto de simulación, es necesario saber identificar los objetivos para los cuales se ha optado por el uso de las técnicas de simulación y formalizarlos de forma que sean precisos, razonables, comprensibles y medibles. Los objetivos deben ser claros y concretos para asegurar que el problema sea abordado correctamente y para que el sistema sea capaz de responder a las expectativas generadas. Estos objetivos servirán de guía a lo largo del proyecto.
- 2. Diseño del modelo conceptual:** una vez conocidos los objetivos del proyecto, es conveniente formular o especificar el modelo de simulación empleando un nivel de abstracción (modelo conceptual) superior al del propio código. El modelo conceptual especifica las relaciones estructurales más importantes del sistema que se intenta simular y, en consecuencia, constituye un medio de diálogo y coordinación entre los distintos grupos involucrados. También corresponde a esta etapa especificar qué resultados o estadísticas se esperan obtener del modelo de simulación para así responder a las preguntas formuladas en la definición de objetivos. Al especificar los resultados, se detecta a menudo qué partes del modelo pueden ser simplificadas o eliminadas dado que no contribuyen a responder dichas preguntas.
- 3. Recopilación y tratamiento de datos:** en general, se recomienda cuestionar siempre toda la información y los datos disponibles: ¿cuál es la fuente? ¿cuándo se obtuvo? ¿cómo se obtuvo? ¿tiene ésta sentido? ¿se tienen suficientes datos o son excesivos? Para obtener buenos resultados es indispensable disponer de buenos datos. Desgraciadamente, en muchos casos no se dispone de ellos. Aun así, se requiere una respuesta a las preguntas planteadas y es necesario efectuar hipótesis razonables en colaboración con el usuario final. Si los datos son limitados o su

Etapas	Descripción
1. Formulación del problema	Define el problema que se pretende estudiar e incluye sus objetivos.
2. Diseño del modelo conceptual	Especificación del modelo a partir de las características de los elementos del sistema que se requiere estudiar y sus interacciones teniendo en cuenta los objetivos del problema.
3. Recopilación de datos	Identificar, recoger y analizar los datos necesarios para el estudio.
4. Construcción del modelo	Construcción del modelo de simulación partiendo del modelo conceptual y de los datos.
5. Verificación y validación	Comprobar que el modelo se comporta como se espera y que existe la correspondencia adecuada entre el sistema real y el modelo.
6. Análisis de resultados	Analizar los resultados de la simulación con la finalidad de detectar problemas y recomendar mejoras o soluciones.
7. Documentación	Documentar el trabajo efectuado.
8. Implementación	Poner en práctica las decisiones efectuadas con el apoyo del estudio de simulación.

Tabla 2.1: Etapas de un proyecto de simulación

calidad es dudosa, es conveniente ser prudente al extraer conclusiones sobre la base de los resultados de la simulación. Aun en los casos en los que hay problemas con los datos, el conocimiento adquirido y los resultados obtenidos en el estudio de simulación aportan información valiosa para la toma de decisiones.

4. **Construcción del modelo:** en numerosos proyectos de simulación el esfuerzo se concentra más en la construcción de un modelo que en la resolución del problema. En esta misma línea, la obtención de un modelo ejecutable se convierte erróneamente en un objetivo prioritario. La motivación dominante debería ser la comprensión del problema y la obtención de soluciones. Para avanzar más rápidamente en la consecución de estos objetivos, es recomendable construir en primer lugar uno o varios modelos simplificados que caractericen las partes más esenciales del sistema de interés.
5. **Verificación y validación:** en el campo de la simulación, la experiencia recomienda suponer que todo modelo es incorrecto excepto que se demuestre lo contrario. Los esfuerzos en dotar a los actuales simuladores de potentes herramientas para facilitar su uso curiosamente han contribuido a uno de los principales peligros de la simulación: olvidarse del mundo real y aceptar sin reparos los resultados del modelo.

Para tener una garantía razonable de que el modelo de simulación representa la realidad, y como consecuencia tomar decisiones estratégicas u operacionales basándose en los resultados, es absolutamente necesario verificar y validar antes el modelo de simulación.

La verificación consiste en comprobar que el modelo se ejecuta correctamente y de acuerdo a lo especificado en el modelo conceptual, es decir, comprobar que las teorías, hipótesis de trabajo así como suposiciones, son correctas. Si el proceso todavía no existe, es necesario contrastar los resultados con expertos del proceso de interés para comprobar si el modelo se comporta tal como ellos esperan.

En general, la validación del modelo suele realizarse comprobando si éste se comporta de manera similar al fenómeno estudiado. Esta es una condición necesaria para la validación pero no suficiente para permitir realizar predicciones.

Los aspectos intangibles, o difícilmente cuantificables del entorno que

condiciona el comportamiento del sistema, así como las características teóricas de los fenómenos subyacentes resultarán críticos en la validación del modelo.

En este caso, el proceso de validación puede separarse en dos pasos. El primero, validación interna, que determina cuándo el modelo es internamente correcto en un sentido lógico. El segundo, validación externa, que determina cuándo el modelo representa el fenómeno que se supone debe imitar.

- 6. Análisis de resultados:** consiste en experimentar con el modelo con el objetivo de efectuar inferencias que permitan tomar decisiones con mayor seguridad. En general, el valor más importante de un estudio de simulación no son los resultados finales obtenidos con el modelo sino el conocimiento adquirido en el proceso de análisis, que permite aportar argumentos cualitativos y cuantitativos justificados a favor o en contra de las diferentes opciones de diseño planteadas.
- 7. Documentación:** es importante mantener un documento al día que refleje el estado del proyecto. Por tanto, el documento evolucionará y se enriquecerá en paralelo con el proyecto de simulación. Los objetivos perseguidos con la elaboración de documentación son:
 - (a) Reflejar el estado del proyecto en un momento dado. De esta forma, todo el personal involucrado en el proyecto tiene información al día sobre su progreso.
 - (b) Informar sobre todo el proyecto (documento final).
 - (c) Facilitar la reutilización del modelo en los casos en los que se prevé un posible interés en su uso futuro. Se recomienda recoger en los documentos la siguiente información: introducción, objetivos, hipótesis, descripción física del sistema, descripción del modelo, análisis de los experimentos efectuados y conclusiones.
- 8. Implementación:** se entiende por implementación el tomar decisiones como consecuencia del estudio de simulación. Un proyecto de simulación no tiene éxito si ha justificado técnica y económicamente una mejora que no se llegue a implementar.

Para obtener mejores resultados de las recomendaciones anteriores, es necesario que el modelo sea creíble; para lograrlo, no solo es suficiente que el modelo sea válido, si no también es imprescindible que los responsables de la toma de decisiones estén convencidos de su validez.

2.9. Los Simuladores

Un modelo, como un conjunto de instrucciones, necesita algún agente que sea capaz de obedecer instrucciones y generar un comportamiento. A este agente se le llama *simulador*. Un simulador es cualquier sistema de cálculo capaz de ejecutar un modelo para generar su comportamiento; es un sistema que se diseña intencionalmente para ser sintetizado a partir de los componentes que no son completamente conocidos y de los que sí son bien conocidos.

Màrques [27] señala que los simuladores representan un modelo o entorno dinámico, generalmente a través de gráficos o animaciones interactivas. Facilitan a los alumnos la exploración y modificación del sistema. Propician un aprendizaje inductivo o deductivo mediante la observación y manipulación del sistema; de esta manera pueden descubrir los elementos del modelo, sus interrelaciones, tomar decisiones y adquirir experiencia ante situaciones que frecuentemente resultarían difícilmente accesibles a la realidad. Por ejemplo, el control de una central nuclear, contracción del tiempo, el pilotaje de un avión, entre otros. También se pueden considerar simulaciones ciertos videojuegos que, al margen de las consideraciones sobre los valores que incorporan, facilitan el desarrollo de los reflejos, la percepción visual y la coordinación psicomotriz, además de estimular la capacidad de interpretación y reacción ante un medio concreto.

Según Cabanellas [5] el objetivo de todo simulador será reproducir cada una de las partes del sistema o proceso de la forma más realista posible, pero siempre manteniendo bajo control la complejidad del sistema modelado. Esta variable es fundamental pues podría hacer crecer de forma desmesurada el desarrollo sin aportar grandes mejoras. Por tanto, es necesario alcanzar un compromiso entre la profundidad del modelado y las funcionalidades que se desean retener de los sistemas reales.

Separando los conceptos de modelo y simulador, se proporcionan una serie de beneficios:

- El mismo modelo, expresado en un formalismo, puede ser ejecutado por diferentes simuladores, de esta manera, ofrece portabilidad e interoperatividad a un alto nivel de abstracción.
- Los algoritmos del simulador pueden ser formulados para varios formalismos y establecer una correctez rigurosa.
- Los recursos requeridos para simular correctamente un modelo permiten medir su complejidad [48].
- Habilitar el estudio y la experimentación con las interacciones de un sistema complejo o de un subsistema dentro de un sistema complejo.
- Los cambios de información, organización y ambiente de un sistema pueden ser simulados, esto permite observar el efecto de tales alteraciones sobre el comportamiento del modelo.
- Para poder sugerir mejoras en el sistema que está bajo investigación, pues el conocimiento adquirido en el diseño de un modelo de simulación puede ser de gran valor.
- Para obtener una valiosa intuición sobre la importancia de las variables y cómo éstas interactúan mediante el cambio de entradas en la simulación y la observación de los resultados de salida.
- Como dispositivo pedagógico para reforzar metodologías de solución analítica.
- Para experimentar con nuevos diseños o políticas antes de la implementación y, de esta manera, prepararse para lo que pueda ocurrir.
- Para verificar soluciones analíticas [8].

Además de las numerosas ventajas que trae consigo el uso de simuladores, gran parte del éxito de los mismos depende del acierto al fijar el compromiso entre profundidad del modelado y la complejidad del proyecto.

2.10. Tipos de simuladores

Los simuladores pueden clasificarse atendiendo a su objetivo en al menos seis categorías: de diseño, de formación, de pruebas, de recreación, de operación y de mantenimiento.

2.10.1. Diseño

Los simuladores de diseño se utilizan para la reproducción de un proceso o de un sistema, con el objetivo de realizar variaciones en los parámetros característicos del modelo y obtener conclusiones sobre el comportamiento del sistema real. Finalmente, el funcionamiento del sistema real se pretende optimizar en base a las conclusiones resultantes de su estudio.

Estos simuladores no presentan requerimientos de tiempo real y la interfaz no necesita reproducir ningún elemento de la realidad, sin embargo, los modelos matemáticos deben tener una complejidad y precisión elevadas.

2.10.2. Formación

Los simuladores de formación tienen por objetivo reproducir un sistema de manera que la interacción con el usuario sea lo más realista posible y son utilizados para entrenar a los usuarios en su relación con el sistema.

El éxito de la simulación para entrenamiento radica en no sólo entrenar la parte consciente del usuario sino engañar o hacer creer al subconsciente del entrenamiento.

El diseño de un simulador para formación deberá garantizar las siguientes ventajas:

- Reducción significativa del tiempo de formación.
- Entrenamientos intensivos en situaciones críticas o de emergencia, hasta su completo dominio.
- Reducción del estrés de formación en máquina real.

- Posibilidad de adaptación de los niveles de formación dependiendo de cada persona.
- Posibilidad de evaluación conjunta con el alumno una vez finalizado el ejercicio.

Los simuladores de formación presentan requerimientos de tiempo real; los modelos matemáticos requeridos deben ser bastante completos para reproducir de forma precisa los elementos del sistema que representa.

2.10.3. Pruebas

En los simuladores de pruebas se trata de reproducir el entorno que rodea a un determinado elemento o sistema, simulando las mismas entradas que éste recibiría cuando se encontrase funcionando en el entorno real. Con estas condiciones se somete al sistema a una serie de pruebas que permitan validar el comportamiento del mismo, con la gran ventaja de realizarlas en un entorno controlado y libre de riesgos.

De esta forma se consigue comprobar la funcionalidad del sistema antes de instalarlo en su emplazamiento final, sometiéndolo a una serie de pruebas amplias y de bajo costo.

Los simuladores de pruebas presentan requerimientos de tiempo real, los modelos matemáticos deben ser muy completos para reproducir fielmente las entradas del equipo y la interfaz no necesita reproducir el aspecto del sistema que representa.

2.10.4. Mantenimiento

Los simuladores de mantenimiento, como indica su nombre, son diseñados para proporcionar ayuda en el mantenimiento de un determinado elemento o de un sistema completo. Se caracterizan por incluir no solo el modelado del elemento sino también información adicional del mismo como manuales de usuario, plan de mantenimiento, entre otros. También pueden aportar información adicional del elemento que permita planificar el mantenimiento

de una forma más sencilla, por ejemplo, dimensiones de las áreas a utilizar a fin de conocer las dimensiones máximas del equipo que se puede emplear.

Este tipo de simuladores constituyen una herramienta integrada de información, bajo una interfaz de usuario totalmente amigable al poder incorporar componentes de realidad virtual.

Los simuladores de mantenimiento no presentan requerimientos de tiempo real y la interfaz debe reproducir al menos una parte del sistema real.

2.10.5. Recreación o reconstrucción

El objetivo de los simuladores de recreación o reconstrucción es intentar reproducir un evento complejo que se ha producido, como por ejemplo, la reconstrucción de accidentes de tráfico.

En general, éstos simuladores deben incorporar modelos matemáticos muy completos al ser los fenómenos que se pretende reproducir de una complejidad elevada.

Este tipo de simuladores no presentan requerimientos de tiempo real y la interfaz no necesita reproducir ningún elemento de la realidad, pero es conveniente en algunos casos para hacerse una idea más intuitiva del resultado de la simulación.

2.10.6. Ocio y entretenimiento

Los simuladores de ocio y entretenimiento tienen el objetivo de entretener o provocar emociones intensas a un amplio espectro de personas. Lo más importante es que la imaginación y sugestión funcionen a pleno rendimiento en el usuario, no importando tanto la fidelidad a la realidad.

Dado que el usuario no tendrá, en una proporción muy elevada, formación ni interés en los fenómenos físicos subyacentes, ni tampoco finalidades formativas, los modelos matemáticos que se emplean son sencillos pero rápidos; lo más complejo es afinarlos para que produzcan el grado de emociones subjetivas que se espera de ellos.

El requerimiento de tiempo real es obvio; la interfaz debe ser lo más realista posible, además, estos simuladores son acompañados por lo general de movimiento, vibraciones y sonidos.

2.11. Motores de física

Un motor de física, también llamado *physics engine*, es un programa que simula modelos físicos Newtonianos utilizando variables tales como masa, velocidad, fricción y resistencia del viento. Puede simular y predecir efectos bajo ciertas condiciones que se aproximan a lo que sucede en el mundo real o ficticio que se desee simular. Su uso principal está enfocado principalmente hacia las simulaciones científicas y los videojuegos. Ejemplos de simuladores de física son: los de código abierto como *Bullet*, *Open Dynamics Engine*; los de código cerrado pero de libre distribución como *Newton Game Dynamics*, *Simple Physics Engine*; entre los de tipo comercial se encuentra *Havok*, que a pesar de ser de uso comercial, tiene una parte llamada *Xtra*, utilizada para el desarrollo de este trabajo, que es de libre distribución.

2.11.1. Funciones de los motores de física

En una simulación, una vez que se han establecido las condiciones iniciales de los componentes del sistema de interés, se inicia el ciclo de simulación, es entonces cuando los motores de física se encargan de desempeñar una serie de tareas específicas que posteriormente deben ser integradas en una salida gráfica para que el usuario pueda visualizar el resultado de la simulación. Estas tareas se dividen principalmente en tres fases:

- 1. Detección de colisiones:** El motor rastrea el movimiento de todos los objetos de la escena y detecta si alguno de ellos ha chocado.
- 2. Actualizar el sistema:** Determina una respuesta apropiada para los objetos que han chocado mediante la resolución de la colisión de acuerdo a las propiedades del objeto y los demás objetos que no chocaron, actualizándolos respecto a las fuerzas que actúan sobre ellos.

3. **Mostrar en pantalla el resultado de la simulación:** Una vez que han sido determinadas las nuevas posiciones de los objetos, es necesario mostrarlos en pantalla para poder visualizar los cambios de estado que han sufrido.

Las tareas anteriores forman parte del *ciclo de una simulación*, que de acuerdo con García [17], se define como el intervalo transcurrido en cada recorrido del ciclo principal del programa. Este intervalo contiene la lectura de los eventos del usuario, gestión del comportamiento de los objetos de la escena, emisión de sonidos, si es el caso, y la visualización de la escena. El tiempo empleado por un ciclo de simulación corresponde al periodo de muestreo de un simulador continuo en el que la frecuencia de muestreo depende de la potencia de cálculo disponible y la complejidad del sistema a simular.

2.12. Simuladores de física

Son herramientas que permiten realizar simulaciones de fenómenos bajo la influencia de las leyes de la física, reaccionando a factores como gravedad, velocidad del viento, resistencia, fricción, calor, agua, etcétera. En la actualidad existen numerosas herramientas que sirven para simular fenómenos físicos, la mayoría de ellas están enfocadas a la enseñanza de la Física. Dichas herramientas están dirigidas a personas que no son especialistas en desarrollos computacionales pero que podrán manejarse con cierta facilidad; la mayoría de los simuladores de física presentan experimentos en 2D, otros pueden manipularse en tiempo real y algunos de ellos son de libre distribución y están disponibles en internet. Algunos ejemplos de simuladores de física son: *Interactive Physics*, *Crocodile Physics*, *Newton Explore Physics*, *Phun*, entre otros.

2.13. Comentario final

La simulación es una herramienta de análisis cuantitativo que promueve el aprendizaje por descubrimiento. Consiste en construir modelos que descri-

ban la parte esencial del comportamiento de un sistema de interés así como realizar experimentos con el modelo y generar observaciones.

La simulación de entornos virtuales siempre se mueve en los límites de la tecnología y aprovecha toda capacidad de cómputo disponible, el empleo de hardware más rápido, algoritmos más avanzados y depurados, permiten simular entornos con mayor grado de detalles.

Particularmente en el desarrollo de éste capítulo se resalta la importancia del uso de simuladores como una herramienta que puede contribuir a la enseñanza de la física, porque se adapta a las necesidades que se deben cubrir en cada curso y además el sistema puede desarrollarse de manera tal que pueda estudiarse un mismo tema de acuerdo a las características y necesidades de los usuarios.

Capítulo 3

Multimedia y Ambientes 3D

La elaboración de software educativo involucra el estudio de aquellas tecnologías que satisfagan las necesidades de construcción, presentación, visualización e interacción con el sistema, principalmente cuando se trata de integrar diversos componentes del software que han sido creados con distintas herramientas de propósito específico. En consecuencia, la elección de la tecnología más adecuada, dependerá de los objetivos que el sistema deba cumplir.

En el caso del desarrollo de sistemas de apoyo a la enseñanza de temas relacionados con la Mecánica Clásica, como el de este trabajo, es necesario representar los fenómenos físicos de tal manera que el usuario sea capaz de reconocer los elementos que están involucrados dentro de cada experimento, en este sentido, los ambientes 3D constituyen una gran alternativa para el modelado y la construcción de mundos artificiales sencillos tan apegados a la realidad como se desee. Por otro lado, una vez que se tiene el ambiente 3D, es imprescindible dotar de dinámica a los componentes del fenómeno, de tal manera que se logre que éstos se comporten como se esperaría que sucediera en la realidad, es decir, que sea posible realizar una simulación y posteriormente se debe poder controlar el comportamiento de tales objetos, con lo que se obtiene un simulador.

Es evidente que en el proceso de desarrollo del sistema se involucran múltiples herramientas que deben conjuntarse en un solo espacio, por lo tanto, la tecnología multimedia es la idónea para lograrlo porque permite integrar y presentar el ambiente 3D, los dispositivos de visualización del

mismo, los dispositivos de interacción con el sistema así como aquellos que permiten la generación y presentación de videos; de ahí su relevancia en el desarrollo de materiales educativos en múltiples campos.

3.1. Multimedia

En la literatura especializada se pueden encontrar varias definiciones de multimedia que se complementan, por ejemplo, para England [10], Zuluaga [49] y Gándara [16], Multimedia es un sistema que consiste en la integración de diversos medios como texto, hipertexto, audio, imágenes, video y todo tipo de software de control dentro de un solo ambiente digital de información, logrando captar significativamente la atención del usuario y brindándole el control interactivo de la aplicación mediante una computadora.

Una aplicación multimedia permite al usuario realizar búsquedas y navegar a través de la información, general o específica, sin la necesidad de recorrer todo el programa, es decir, la información contenida en la aplicación se maneja de manera asociativa, representando una importante ventaja frente al modelo pasivo que caracteriza a los medios tradicionales de comunicación.

Esta definición aplica tanto a los medios producidos para su distribución *en línea (online)* como las páginas web; a los medios producidos para su distribución *fuera de línea (offline)*, es decir, que no necesitan de alguna conexión a Internet, como las aplicaciones contenidas en CD-ROM, o bien, aplica a los proyectos *híbridos web/CD* que son aquellos que tienen elementos a los que se accede tanto *en línea* como *fuera de ésta*.

Un proyecto *fuera de línea* es por definición autocontenido, es decir, no interactúa con nada fuera de su ambiente inmediato y por lo general no se comunica más que con la computadora en la que se está ejecutando a excepción del usuario. En cambio, un proyecto *en línea*, necesita comunicarse con recursos o usuarios distantes, o bien, ejecutarse desde una computadora distante.

Cabe aclarar que un proyecto multimedia, independientemente de si está en línea, fuera de línea o es híbrido, agrega Zambrano [46], no tiene que ser estrictamente interactivo para llamarse multimedia: existen aplicaciones

en las que los usuarios pueden reclinarsse en el asiento y verlas como lo hacen en el cine o frente a la televisión. En tales casos se dice que un proyecto es lineal, pues empieza y se ejecuta hasta el final, en contraste, un sistema es interactivo cuando se da el control de navegación a los usuarios para que exploren el contenido de la aplicación.

Existen dos grupos principales de materiales que se pueden integrar dentro de un mismo sistema multimedia: textuales y audiovisuales. Los materiales textuales pueden ser secuenciales, hipertextuales o una combinación de ambos; por otro lado, los materiales audiovisuales pueden ser gráficos, sonidos, videos, animaciones entre otros.

Las aplicaciones multimedia comprenden productos y servicios que van desde la computadora y sus dispositivos especiales para las tareas multimedia como bocinas, pantallas, unidades lectoras de dispositivos extraíbles hasta las comunicaciones virtuales que posibilita Internet, pasando por los servicios de video interactivo en una televisión y las videoconferencias.

Cuando un programa de computadora, documento o presentación combina adecuadamente los medios, facilita la comprensión y el aprendizaje, pues se acercará un poco más a la manera habitual de comunicación humana, mediante el empleo de varios sentidos para entender un mismo concepto, logrando mejorar notablemente la atención y realzar la experiencia del usuario.

Los programas multimedia, por su alto contenido de información, deben ser distribuidos en medios de gran capacidad que hagan práctica su utilización. El CD-ROM se ajusta a estas características y además resulta ser un medio muy económico.

Recientemente, con el vertiginoso desarrollo de Internet, se ha abierto una puerta muy importante para la distribución de las aplicaciones multimedia, sin embargo, las velocidades de transmisión de datos la hacen poco práctica por los tiempos de respuesta. No obstante se espera que en el futuro sea el medio de distribución más popular aunque, como señala Alonso [1], es importante que tanto alumnos como profesores no sólo aprendan a tener acceso a la información a través de Internet sino a manejarla, analizarla, criticarla, verificarla y transformarla en conocimiento utilizable. Deben poder escoger lo que realmente es importante, dejando de lado lo que no lo es.

Algunas de las áreas de aplicación de los sistemas multimedia son el arte,

la educación, el entretenimiento, la ingeniería, la medicina, las matemáticas, los negocios y la investigación científica.

3.2. Los sistemas multimedia en la educación

Multimedia es una tecnología utilizada frecuentemente para la producción de software educativo aunque, comenta Zambrano [47], multimedia no puede resolver completamente ni directamente los problemas que impone la transmisión de conocimientos dentro del proceso de enseñanza-aprendizaje, pero en cambio, sí posee los elementos que intervienen adecuadamente en la construcción de soluciones para simular de manera más real, las funcionalidades pedagógicas que le son necesarias al docente.

Los sistemas multimedia, como sistemas concebidos para la comunicación, buscan realzar y enfatizar conceptos, alentando al estudiante a participar e involucrarse en su propio proceso de enseñanza, en el que jugará un rol activo gracias a las facilidades que presentan las aplicaciones multimedia, despertando la participación del alumno y adaptándose a su forma de aprender, así, durante su interacción con el sistema, los estudiantes trabajan sin estar sometidos a la presión que representan una velocidad y un ritmo de trabajo preestablecidos.

Son muchos los recursos tecnológicos relacionados con multimedia que pueden integrarse en la elaboración de software educativo y de herramientas didácticas al alcance del docente, proporcionándole los medios necesarios para que el alumno pueda aprender viendo, oyendo y haciendo, a través de un modelo simulado. El problema en tal integración reside en la poca reflexión hecha sobre las funcionalidades pedagógicas de base existentes y sus correspondientes traducciones en términos informáticos.

Un sistema multimedia interactivo brinda el sustento tecnológico necesario para un aprendizaje efectivo. La posibilidad de dividir la pantalla en partes permite al usuario comparar selectivamente imágenes fijas o en movimiento, estudiar cualquier cosa, desde la preparación de alimentos hasta la reparación de un automóvil, por mencionar algunos. Los gráficos de alta calidad pueden ser combinados con video para permitir al estudiante ver y escuchar la explicación de un procedimiento y luego practicar en otra sec-

ción de la pantalla. Múltiples eventos pueden ocurrir simultáneamente en un orden al azar para simular eventos complejos que requieren entrenamiento.

La incorporación de multimedia a la educación, afirma Gándara [16], tiene sentido dado que se cumplen las siguientes afirmaciones:

- Los multimedios permiten una comunicación rica, al impactar varios canales de percepción, incluyendo el kinestésico o de movimiento, lo que permite que, la presentación se adecue al estilo de aprendizaje del usuario.
- Su naturaleza interactiva, combinada con la alta capacidad de los dispositivos de almacenamiento digital, como el CD-ROM, permite que pueda variarse el orden y la profundidad de la presentación de la información de manera jerarquizada, adecuándose así a las necesidades de estudiantes con diferentes niveles de familiarización con la información presentada.
- Los multimedios permiten narrativas ricas, con elementos dramáticos y lúdicos que permiten “aprender jugando”, afectando no solamente en el ámbito cognitivo, sino en el emocional.
- Finalmente, es la interactividad la que, mediante recursos como la simulación, con la creación de modelos en computadora que el usuario manipula y modifica, permite que este aprenda haciendo.

Estas características de los multimedios se combinan con las que, de por sí, tiene ya la computadora.

- La capacidad de ofrecer una respuesta inmediata y oportuna al usuario.
- De tener una infinita paciencia y consistencia en la entrega de la instrucción.
- De almacenar las respuestas del aprendiz y trazar su trayectoria y avance.
- De permitir acceder a conjuntos grandes de información, tanto de manera local como remota.

- De enlazar a aprendices y maestros en una enorme comunidad internacional de aprendizaje.

En conjunto, estas son características de los sistemas de cómputo, particularmente de los sistemas multimedia, las que un buen diseño instruccional debe potenciar para un buen aprovechamiento de este medio. De ahí también que algunas de las experiencias de aprendizaje que puede ofrecer la computadora son únicas, o al menos, mucho más fáciles de realizar en la computadora que sin ella. Son este conjunto de capacidades las que justifican su uso como nuevas herramientas educativas.

3.3. Clasificación de los sistemas multimedia educativos

Las aplicaciones multimedia interactivas, indica Belloch [4], se clasifican en función de tres diferentes criterios: según su sistema de navegación, según su finalidad y base teórica y según el nivel de control que tiene el profesional sobre la aplicación.

3.3.1. Clasificación según su sistema de navegación

La estructura seguida en una aplicación multimedia es de gran relevancia pues determina el grado de interactividad de la aplicación, por lo tanto, la selección de un determinado tipo de estructura para la aplicación condicionará el sistema de navegación seguido por el usuario y la posibilidad de una mayor o menor interacción con la aplicación. No existe una estructura mejor que otra, la selección de la misma dependerá del objetivo de la aplicación multimedia.

Los sistemas de navegación más usuales en relación a la estructura de las aplicaciones son: lineal, reticular y jerarquizado.

- **LINEAL:** El usuario sigue un sistema de navegación lineal o secuencial para acceder a los diferentes módulos de la aplicación, de tal modo

3.3. CLASIFICACIÓN DE LOS SISTEMAS MULTIMEDIA EDUCATIVOS 59

que únicamente puede seguir un determinado camino o recorrido. Esta estructura es utilizada en la mayoría de las aplicaciones multimedia de ejercitación y práctica, así como en los libros multimedia.

- **RETICULAR** Se utiliza el hipertexto para permitir que el usuario tenga total libertad para seguir diferentes caminos cuando navega por el programa, atendiendo a sus necesidades, deseos, conocimientos, etcétera. Es la estructura más adecuada para las aplicaciones orientadas a la consulta de información, por ejemplo, para la realización de una enciclopedia electrónica.
- **JERARQUIZADO:** Combina las modalidades lineal y reticular. Este sistema es muy utilizado pues combina las ventajas de los dos sistemas anteriores: libertad de navegación y organización de la información atendiendo a su contenido, dificultad, etcétera.

3.3.2. Clasificación según su finalidad y base teórica

Se han desarrollado gran cantidad de aplicaciones multimedia con diferentes objetivos y funciones pedagógicas, por ejemplo, enciclopedias multimedia, cuentos interactivos, juegos educativos, tutoriales, entre otros. La finalidad de estas aplicaciones puede ser de dos tipos: predominantemente informativa o formativa.

- **Multimedia informativo:**
 - **Libros o cuentos multimedia.** Se parecen a los libros convencionales porque mantienen una estructura lineal para el acceso a la información, pero en su contenido tiene mayor importancia el uso de sonidos y animaciones para la presentación de esta información.
 - **Enciclopedias y diccionarios multimedia.** Al igual que las enciclopedias y diccionarios en papel, son recursos de consulta de información, por lo que su estructura es principalmente reticular para favorecer el rápido acceso a la información. Utilizan bases de datos para almacenar la información de consulta de forma estructurada, de modo que el acceso a la misma sea lo más rápido y sencillo posible.

- **Hipermedias.** Son documentos con información relacionada a través de enlaces que presentan información multimedia. Su estructura es en mayor o menor grado jerarquizada utilizando diferentes niveles de información. No obstante, los usuarios tienen gran libertad para moverse dentro de la aplicación atendiendo a sus intereses.
- **Multimedia formativo:**
- **Programas de ejercitación y práctica.** Presentan un conjunto de ejercicios que deben realizarse siguiendo la secuencia predefinida del programa. Utilizan una retroalimentación externa para el refuerzo de las actividades. Tienen un importante uso en actividades que exigen el desarrollo y ejercitación de destrezas concretas.
 - **Tutoriales.** Son semejantes a los programas de ejercitación pero presentan la información que debe conocerse o asimilarse previamente a la realización de los ejercicios. En muchos tutoriales se representa la figura del tutor mediante una imagen animada o un video que va guiando el proceso de aprendizaje.
 - **Simulaciones.** Permiten al usuario experimentar con gran variedad de situaciones reales. Básicamente el programa muestra un escenario o modelo sobre el que el estudiante puede experimentar, ya sea indicando determinados valores para las variables del modelo, o bien, realizando determinadas acciones sobre el mismo, comprobando a continuación los efectos que sus decisiones han tenido sobre el modelo propuesto. De este modo, el usuario toma un papel activo en su proceso de aprendizaje, decidiendo qué hacer y analizando las consecuencias de sus decisiones.
 - **Talleres creativos.** Promueven la construcción de nuevos entornos creativos a través del uso de elementos simples. Por ejemplo, juegos de construcción, talleres de dibujo, entre otros.
 - **Resolución de problemas.** Estas aplicaciones multimedia tienen el objetivo de desarrollar habilidades y destrezas de nivel superior, planteando problemas contextualizados en situaciones reales que requieren el desarrollo de destrezas tales como comprensión, análisis, síntesis, etcétera. Proporcionan materiales y recursos

3.3. CLASIFICACIÓN DE LOS SISTEMAS MULTIMEDIA EDUCATIVOS 61

para su solución, junto a materiales adicionales para profundizar el tema planteado.

- **Caza del tesoro.** Es una página web en la que se presentan una serie de preguntas sobre un determinado tema, junto a una lista de direcciones web en las que se pueden buscar las respuestas. Como punto final se incluye una pregunta: “la gran pregunta”, que los alumnos deben responder a partir de la comprensión e integración de lo aprendido durante la búsqueda y resolución de las preguntas, pues no es posible encontrar la respuesta de forma directa.
- **WebQuest.** Es una actividad orientada a la investigación, en la que parte o toda la información con la que interactúan los alumnos, proviene de Internet; usa el mundo real y tareas auténticas para motivar a los alumnos. Consiste en seis partes esenciales: Introducción, Tarea, Proceso, Recursos, Evaluación y Conclusión; por tanto motiva a los alumnos a transformar la información y entenderla.
- **Wiki.** Básicamente consiste en la elaboración de documentos multimedia de forma colaborativa. Los documentos, llamados páginas wiki se alojan en un servidor y pueden ser escritos por un conjunto de personas a través de un navegador, utilizando una notación sencilla para dar formato, crear enlaces, etcétera. Cuando alguien edita una página wiki, sus cambios aparecen inmediatamente en la web, sin pasar por ningún tipo de revisión previa.

3.3.3. Clasificación según el nivel de control que tiene el desarrollador

Una de las características más deseables en una aplicación multimedia es su capacidad para poder ser configurado y adaptado por el profesional para poder atender las necesidades concretas de los usuarios. Los tipos de software según el menor o mayor nivel de control por parte del profesional son:

- **Programas cerrados.** Lo componen los programas que trabajan sobre un determinado contenido, y el profesional no tiene posibilidad de modificarlo o adaptarlo a las características de las personas

con las que trabaja. Tienen una estructura secuencial que no puede ser modificada por el usuario.

- **Programas semiabiertos.** Estas aplicaciones permiten que el profesional modifique algunas características del programa, por ejemplo, seleccionar diferentes niveles de dificultad en las actividades a realizar, así como adaptar la interfaz de usuario a las características y necesidades del mismo.
- **Programas abiertos.** Son programas que, partiendo de un conjunto de posibilidades de actuación, permiten que el profesional fije el contenido concreto a desarrollar, pudiendo adaptarlo a necesidades concretas del usuario.

En general puede observarse que una computadora personal multimedia brinda todos los niveles de acceso a la administración de un gran conjunto de recursos, representando así una gran ventaja para el desarrollo de sistemas multimedia orientados a la educación.

3.4. Ambientes 3D

Los ambientes 3D son aquellos que permiten representar (modelar) un sistema replicando aspectos suficientes de la realidad que serán regidos por una serie de reglas, no todas estrictamente iguales a las de la vida real (simulación).

Cuando se utiliza un ambiente 3D para modelar un fenómeno físico, es posible que el usuario tenga control del sistema creado e interactúe manipulando los objetos y sus parámetros de modo tal que esos cambios produzcan una reacción en el mundo artificial donde además, podrá observar los objetos del sistema a través de una serie de perspectivas que le transmitirán una sensación de volumen y profundidad, contribuyendo a la visualización del fenómeno en estudio de manera más apegada a como se vería en la realidad.

3.5. Software multimedia para la creación de sistemas 3D

El software es el componente clave para cualquier sistema; permite al usuario crear un ambiente 3D proporcionando las interacciones y visualizaciones necesarias para su funcionamiento. Es probable que durante el desarrollo de un sistema se empleen distintos tipos de software dependiendo de la etapa de desarrollo, por lo tanto, es importante entender los diferentes roles de este software, pues varían enormemente en términos de funcionalidad, utilidad y facilidad de uso.

En el diseño de ambientes tridimensionales es muy importante que el software empleado para su construcción ofrezca capacidades de navegación en tiempo real, interacción con el usuario, detección de colisiones, audio, programación de comportamientos, etcétera. De este modo en los sistemas 3D se encontrarán comúnmente las siguientes características [32]:

1. Importación de modelos: es la capacidad de importar formas tridimensionales para incorporarlas a una determinada aplicación.
2. Bibliotecas: la mayoría de los sistemas 3D están provistos de bibliotecas con formas básicas o primitivas tridimensionales tales como cajas, esferas, conos, pirámides, entre otras, que sirven para generar formas compuestas. También cuentan con bibliotecas de objetos complejos, texturas y elementos de iluminación, permitiendo que el diseñador las reutilice y las adapte a sus necesidades.
3. Operaciones geométricas: consiste en las capacidades de manipular objetos creados en una posición definida, típicamente referida a las coordenadas cartesianas, a partir de las cuales es posible realizar transformaciones que modifiquen la posición, la orientación y el tamaño de los objetos. Se incluyen operaciones booleanas y agrupamiento de formas para crear objetos operando y asociando distintas formas simples.
4. Nivel de detalle: permite la optimización de la visualización de una escena virtual al cambiar una forma con un alto nivel de detalle por otra más simple dependiendo de la distancia del punto de vista. De este modo el objeto es reemplazado o se hace invisible si el observador

está en movimiento o muy distante, y cuando el usuario está más cerca, se despliega la forma más compleja.

5. Animación: corresponde a la asignación de una traslación o rotación de un objeto en determinado periodo, sincronizado con la navegación por el ambiente 3D. Estos movimientos generalmente equivalen a comportamientos del mundo real que pueden ser automáticos o generados por algún evento, es decir, por la interacción con el usuario o con otro objeto.
6. Articulado: se refiere a que los objetos pueden ser originados en jerarquías, es decir, que partes componentes de un objeto posean propiedades de movimiento distintas a otras partes pero que estén supeditadas al total.
7. Detección de colisiones: es una característica que permite identificar cuando un objeto interfecta a otros, de modo que pueda ser obstaculizado el movimiento del objeto, suponiendo que tales objetos son sólidos como sucede en el mundo real.
8. Propiedades físicas: adicionalmente se presenta una serie de atributos relacionados con características físicas tales como masa, fricción, reconocimiento de gravedad, etcétera.
9. Color y texturas: consiste en la asignación de colores a las superficies y al uso de texturas digitalizadas, incluyendo propiedades de transparencia y secuencias o videos.
10. Fuentes de luz: es la definición de la iluminación ambiental y focos de luz con cierta posición, orientación intensidad, e incluso colores y movimientos propios.
11. Incorporación de audio: consiste en la asociación de sonidos a los objetos, tal como corresponde en el mundo real.
12. Lenguajes de programación: el software dispone de comandos de control que dicten el comportamiento de los objetos y manejen datos de entradas y salidas.

3.5. SOFTWARE MULTIMEDIA PARA LA CREACIÓN DE SISTEMAS 3D65

13. Manipulación de eventos: refleja la capacidad de activar un comportamiento al interactuar con un determinado objeto, esto implica reconocer la posición o acción del usuario y modificar la geometría como consecuencia de dicha manipulación.
14. Configuración de múltiples dispositivos: consiste en permitir la incorporación de distintos dispositivos de entrada y salida de datos como elementos de visualización o de interacción con el usuario.
15. Mundos paralelos: generación de ambientes virtuales constituidos por submundos en los que el navegante puede interactuar al momento que ingresa en cada uno de ellos con el fin de disminuir la complejidad que se tendría si fuera un solo mundo completo.
16. Conectividad en red: permite que el mundo virtual sea utilizado en una red computacional a través de diversos dispositivos o señales de entrada y salida, permitiendo la interacción con diversos usuarios en una misma aplicación.
17. Exportación VRML: consiste en exportar las aplicaciones virtuales en lenguaje VRML que pueden ser utilizadas en Internet.

El software utilizado para generar sistemas 3D depende principalmente del papel que éste deba desempeñar de acuerdo a la etapa de desarrollo del sistema, pudiendo utilizar más de un tipo de software a la vez. Se divide en las siguientes categorías: Software de modelado y definición de objetos 3D (geometría, textura, comportamiento, etcétera), software generador de ambientes virtuales, software de diseño y software de autoría.

3.5.1. Software de modelado

Un ambiente 3D comprende una colección de objetos que se caracterizan por su geometría; la apariencia de su superficie, es decir, tanto su textura como su tipo de material; y por la representación de su comportamiento, es decir, todos los datos referentes a su física, dinámica y las propiedades del material.

Un ambiente 3D está creado por la combinación de una colección de objetos ligados a varios comportamientos. El software que realiza la función

de crear las bases de datos de los objetos, es referenciado por el sistema de modelado. En varios aspectos, la calidad visual del ambiente 3D está en función de la capacidad de la herramienta de modelado; es posible utilizar herramientas CAD, pero generalmente no están optimizadas para la creación de objetos para su uso en tiempo real. Por otro lado, un requerimiento clave que deben cumplir los sistemas de modelado, es la capacidad de importar y exportar modelos a otros estándares CAD. Ejemplos de estos sistemas son: 3D Studio MAX, MAYA, Rhinoceros, AutoCAD, entre otros.

3.5.2. Software generador de escenarios sintéticos

Una vez que ha sido creada una colección de objetos 3D, es necesario ligarlos para crear un escenario. Los objetos deben ligarse de acuerdo a la relación que exista entre cada uno de ellos y su interacción con el usuario. Por ejemplo, el programador debe saber qué sucede si el usuario mueve un objeto y luego lo deja caer, ¿caerá bajo la influencia de la gravedad o permanecerá suspendido en el espacio? A los sistemas que permiten verificar tales propiedades de los objetos se les conoce como *motores de física o physics engine*. Es muy importante hacer un buen manejo de restricciones, ya que éstas aseguran que factores tales como la estructura mecánica de los objetos no provoquen problemas que atenten contra el buen funcionamiento del sistema, principalmente cuando se trata de simulaciones. Ejemplos de estos sistemas son: Reactor, Havok, Open Dynamics Engine, entre otros.

3.5.3. Software de diseño

Son programas que sirven para realizar el diseño de los elementos que constituyen la interfaz del sistema; algunos de ellos permiten modelar objetos tridimensionales, editar y generar animaciones para su uso dentro del sistema e incluso proporcionan facilidades para crear recorridos en formato de película digital. Algunos ejemplos son: Photoshop, 3D Studio MAX, Flash, AutoCAD, CorelDraw, entre otros.

3.5.4. Software de autoría

Actualmente, el software de autoría, *authoring software*, permite desarrollar sistemas multimedia facilitando la tarea de los desarrolladores; ofrece las herramientas necesarias para integrar dentro de un mismo sistema elementos tales como texto, audio, imágenes, objetos bidimensionales o tridimensionales, videos, animaciones y simulaciones así como generar sus respectivas interfaces; además, se complementan con un lenguaje de programación propio que controla todo en conjunto. Una de las propiedades importantes del software de autoría es que permite manejar elementos importados desde los sistemas de modelado, de diseño y los generadores de ambientes virtuales. Ejemplos de estos sistemas son: Director, que es el sistema utilizado en el desarrollo del caso práctico de esta tesis, Authorware, ToolBook, IconAuthor, Visual Basic, entre otros.

La mayoría de las herramientas de autoría, como Director, se relacionan con materiales SDK - *Software Development Kit* o herramientas de desarrollo de software, que se trata de un conjunto de herramientas de desarrollo que le permite al programador crear aplicaciones para un sistema en concreto; permite el uso de determinado lenguaje de programación o puede también, incluir hardware sofisticado para comunicarse con algún sistema embebido. Las herramientas más comunes incluyen soporte para la detección de errores de programación y otras utilidades, además de incluir códigos de ejemplo y notas técnicas de soporte.

3.6. Creando ambientes 3D

El proceso de creación de un ambiente 3D puede convertirse en una tarea extremadamente compleja; se divide principalmente en cinco etapas que se explican en la Tabla 3.1 [23], pero sin lugar a dudas, el primer paso para crear un ambiente 3D consiste en definir qué es lo que el usuario debe ser capaz de hacer dentro del ambiente.

Existen distintas maneras de crear o modelar el ambiente 3D que al principio parecerán las más adecuadas. Un error común es modelar todo el ambiente como un modelo único; esto puede ser bueno para tareas de visualización simples, pero después será imposible mover elementos individuales dentro del

ambiente. Este método de construcción puede dificultar la optimización del proceso de rendero, principalmente para interacciones en tiempo real.

Es ideal definir el ambiente 3D como un conjunto de objetos jerárquicamente ligados. Esto no sólo hace más fácil cualquier tarea de mantenimiento del ambiente sino que permite alcanzar un rendero óptimo de los objetos que están visibles en la escena. Este proceso requiere ser más pensado en la fase de modelado, y con ello alcanzar una considerable reducción en el tiempo de modelado. Por otro lado, si el sistema debe ejecutarse en diferentes plataformas, entonces el diseñador tiene una tarea más difícil, pues debe considerar las complicaciones de compatibilidad de plataformas.

3.7. Comentario Final

Tras la facilidad de que la mayoría de las personas tengan acceso a sistemas que no sólo sean de entretenimiento sino también educativos, la gran variedad de tecnologías para crear ambientes virtuales se ha incrementado dramáticamente así como también han sufrido una evolución drástica. En solo pocas décadas, las capacidades de hardware y software para la creación de ambientes tridimensionales han aumentado y con ello, tales ambientes pasaron de ser representaciones simples a representaciones tan complejas que logran engañar la percepción visual, aunque para los fines de éste trabajo, únicamente será necesario presentar formas básicas dentro del ambiente 3D.

Desafortunadamente, las tecnologías de interacción con el usuario no han alcanzado el mismo nivel de desarrollo, no logrando adaptarse completamente a los requerimientos de cada sistema.

El desarrollo de éste capítulo ha sido muy significativo en la implementación del presente trabajo respecto a la toma de decisiones referentes al tipo de herramientas a utilizar para generar el ambiente 3D, así como las ventajas y limitaciones que se deben enfrentar en cuanto a la disposición de tecnologías tanto de despliegue como de manipulación del simulador.

Tarea	Subtarea
1. Definición de la escena	Definición de los objetos 3D Nivel de detalle de los modelos Definición de materiales Descripción de las texturas que serán aplicadas a los modelos Definición de desempeño (modelos altos en polígonos vs. bajos en polígonos, conteo de mapeos de textura) Descripción de la iluminación de la escena Definición de efectos ambientales Definición de interacción entre objetos individuales (comportamiento autónomo o inicializado por el usuario)
2. Modelado de objetos	Modelado geométrico (creación de modelos con diferentes niveles de detalle si es necesario) Aplicación de materiales Aplicación de mapeos de textura
3. Conversión de base de datos (optativo)	Escala de objetos Reducción de polígonos
4. Creación del escenario	Generar la escena (colocar los objetos en el ambiente 3D) Integración del nivel de detalle de los objetos Implementación de la iluminación Implementación del comportamiento autónomo de los objetos Implementación del comportamiento inicializado por el usuario Ligado de cinemática y variables físicas Integración con otras modalidades (audio, video, etcétera)
5. Pruebas	Pruebas de desempeño Optimización Pruebas de interfaz de usuario

Tabla 3.1: Etapas de creación de ambientes 3D

Capítulo 4

Desarrollo de un Sistema de Manipulación Gráfica de Fenómenos Físicos Reales, Asistidos por Computadora

La ciencia es un proceso de conocimiento a través de la observación y el razonamiento; sin embargo, es una actividad que por lo general es vista por los alumnos como un conjunto de problemas teóricos a los que se les debe aplicar una *fórmula* o ecuación determinada para resolverse, pero en numerosas ocasiones, los alumnos nunca llegan a entender el por qué del comportamiento de algún sistema; por lo tanto, es evidente la necesidad de que los alumnos tomen un papel activo e investigador en su proceso de aprendizaje.

En la actualidad el acceso a tecnologías más económicas y de fácil adquisición, posibilita el desarrollo de herramientas especializadas que logran explicar la ciencia desde una perspectiva experimental; de esta manera, se propicia la participación activa y autónoma de los alumnos en temas específicos; entre dichas tecnologías se citan a los simuladores, los ambientes 3D y las aplicaciones multimedia.

Por un lado, los simuladores promueven un aprendizaje por descubrimiento porque permiten representar fenómenos físicos que pueden ser modificados;

facilitan un estudio detallado, reducen riesgos, costos, errores y la estricta necesidad de utilizar instalaciones especiales. Por otro lado, los ambientes 3D proporcionan los elementos para poder llevar a cabo las simulaciones en un ambiente gráfico. Finalmente, la tecnología multimedia posibilita la elaboración de software educativo que permite integrar simulaciones interactivas en una sola herramienta.

Este trabajo se centra en la importancia de la enseñanza de la ciencia, en particular de la Física, porque una de las dificultades a las que se enfrentan los alumnos durante su estudio es que no siempre logran asociar los conceptos analizados desde una perspectiva teórica con los sucesos experimentales de un mismo fenómeno, y la Física es fundamentalmente una ciencia que se basa tanto en la medición como en la analogía entre la realidad y el experimento.

4.1. Antecedentes del proyecto

En México, numerosas instituciones educativas cuentan con laboratorios de cómputo disponibles para el apoyo de los alumnos, no sólo para que éstos utilicen procesadores de texto y hojas de cálculo para cumplir con sus tareas, sino para que además puedan reforzar los temas vistos en clase con la ayuda de software educativo.

A pesar de la gran cantidad de software con fines educativos que existe para apoyar a los alumnos en áreas específicas, la mayoría de estas aplicaciones, presentan uno o varios de los siguientes inconvenientes:

- Están elaborados en idiomas distintos al español.
- Son costosos.
- Son de difícil adquisición.
- No relacionan al alumno con problemas o experiencias que le sean cotidianos.
- No permiten al alumno ligar los aspectos teóricos del conocimiento con los aspectos prácticos del mismo.

Por lo tanto, a raíz de esta problemática y con la finalidad de minimizar dichos inconvenientes, en el Laboratorio de Interacción Humano-Máquina y Multimedia (LIHMyM) surge la idea de desarrollar una serie de materiales interactivos que logren englobar los aspectos teóricos y prácticos de un curso de Física a nivel bachillerato: la *Enciclopedia de la Física*, el *Laboratorio Basado en Video* y finalmente, el *Sistema de Manipulación Gráfica de Fenómenos Físicos Reales, Asistidos por Computadora*. El primero de estos materiales se encarga de cubrir los aspectos teóricos del curso, mientras que los aspectos prácticos son cubiertos con los dos últimos.

4.1.1. La Enciclopedia de la Física

También llamada *Hipertexto de Física* [34], es una herramienta multimedia inspirada en las enciclopedias impresas, salvo que en esta, los alumnos acceden a la información de un modo dinámico e interactivo.

La información contenida en la enciclopedia está dividida en diez asignaturas principales: Mecánica y Fluidos, Ondas, Termodinámica, Óptica, Electricidad, Magnetismo, Física Moderna, Electrónica, Tecnología y el Método Experimental. Además, la información contenida en cada una de las asignaturas, está organizada en cinco categorías: teoría, teoría avanzada, biografías, documentos históricos y experiencias; e integra diversos medios como fotografías, imágenes, videos, representaciones virtuales, simulaciones, gráficas, etcétera.

La *Enciclopedia de la Física* es, en resumen, una alternativa al libro de texto, con la ventaja de ofrecer notas filosóficas para explicar el sentido de los conceptos estudiados en su contexto histórico particular y para entender la manera en que los conceptos físicos están ligados al mundo cotidiano.

4.1.2. El Laboratorio Basado en Video (LBV)

Es una herramienta que permite observar, medir y analizar fenómenos físicos reales que han sido filmados en video formato AVI¹, véase Figura 4.1.

¹AVI es el acrónimo de Audio Video Interleave: Audio y video entrelazado.

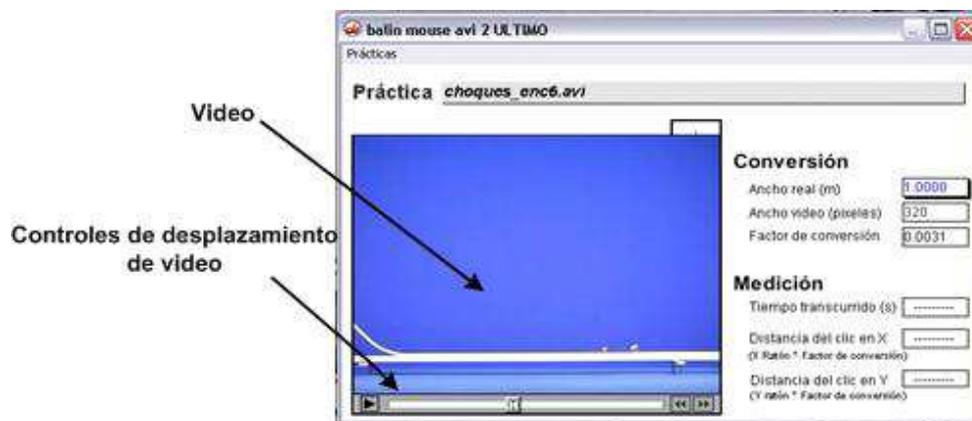


Figura 4.1: Laboratorio Basado en Video

El funcionamiento del LBV [14] es básicamente el siguiente:

- Una vez abierto un video, es posible desplazarlo cuadro por cuadro.
- Se posiciona el ratón sobre algún objeto en movimiento del experimento, y al presionarlo, se obtienen las coordenadas (x, y) del objeto. Repitiendo esta operación, se puede saber la variación de la posición de los objetos a lo largo del tiempo y con estos datos se puede obtener la descripción general del comportamiento del sistema analizado, es decir, el tipo de movimiento correspondiente a cada experimento: movimiento rectilíneo uniforme, uniformemente acelerado, circular, etcétera.

El LBV, además de permitir analizar detenidamente los fenómenos de estudio, facilita la obtención de datos virtuales experimentales muy similares a los datos que se podrían obtener en el laboratorio escolar. La Enciclopedia de la Física y el Laboratorio Basado en Video, son herramientas que los alumnos han aceptado con gran entusiasmo pero, para lograr que éstos relacionaran mejor la teoría con la práctica, era necesario desarrollar una herramienta que los llevara más allá de la sola observación y medición de los fenómenos físicos, esta herramienta debía tener la característica principal de permitir a los alumnos experimentar.

4.2. Comparación del sistema de manipulación gráfica con otros simuladores de física

Actualmente existen numerosos simuladores que permiten estudiar fenómenos físicos. La mayoría de estas herramientas varía en costos, interacción, los temas abarcados, algunos presentan los experimentos en 2D otros en 3D, también están los que permiten al usuario generar sus propios ambientes, etcétera. A continuación se describirán brevemente algunos de los simuladores de física para poder hacer una comparación con el *sistema de manipulación gráfica*.

Interactive Physics: Este software es desarrollado por MSC Software Corp. [22], posee una amplia selección de controles, parámetros, objetos, ambientes y componentes; permite que tanto profesores como estudiantes simulen diversas situaciones físicas permitiéndoles agregar objetos, resortes, articulaciones, sogas y amortiguadores para que diseñen libremente cualquier situación física y compararla con lo que sucede en la realidad. Simula el contacto, las colisiones, la fricción, la gravedad y la resistencia del aire ofreciendo una respuesta gráfica en tiempo real. Incorpora una biblioteca de más de 150 simulaciones listas para usar y estas pueden ser modificadas para resolver necesidades específicas. Todos los experimentos se visualizan en 2D; está disponible en diversos idiomas incluyendo el español y es de paga.

Newton Explore Physics: Es una aplicación que permite realizar experimentos relacionados con cinemática y dinámica en 3D [30]. Al crear un experimento es posible seleccionar las geometrías que conformarán el mundo y ajustar sus parámetros físicos como masa, elasticidad, fricción; asignación de fuerzas, torques o velocidad. Es posible agregar objetos que hayan sido construidos en formato VRML; contiene archivos ejecutables con ejemplos cuyos parámetros pueden ser modificados por el usuario, quien además podrá agregar una o más cámaras y capturar las vistas de los experimentos en formato AVI. Dentro de la herramienta se despliegan una serie de diagramas que ayudan a medir y evaluar los resultados de la simulación así como comparar las gráficas de las curvas descritas por la simulación con las que son definidas por el usuario. El idioma de la aplicación es el inglés y es de paga.

Crocodile Physics: Es un simulador que permite generar modelos relacionados con la electricidad, movimiento y fuerzas, óptica y ondas [7]; permite generar simulaciones mediante el arrastre de las partes que compondrán el sistema en la posición deseada; los parámetros de los experimentos podrán ser modificados, también ofrece la posibilidad de graficar los datos de los experimentos e incluye varias lecciones pre-diseñadas. Los experimentos están en 2D, en idioma inglés y es de

Phun: Es una aplicación gratuita basada en C++ y OpenGL desarrollada en la Universidad de Umeå en Suecia [36], actualmente está disponible para los sistemas operativos Windows y Linux; permite al usuario generar sus propios experimentos en 2D mediante una paleta de herramientas que permite dibujar a modo de caricatura un mundo y asignarle propiedades físicas en tiempo real, el idioma de la herramienta es inglés y está disponible en Internet.

Java physics applets: Los *applets* de Java son una poderosa herramienta que ha permitido que varias universidades del mundo construyan sus propios Laboratorios Virtuales de Física y de esta manera apoyar el proceso de aprendizaje en esta ciencia cubriendo diversos temas como mecánica, termodinámica, astrofísica, etcétera. La mayoría de estos experimentos está en 2D y el estudiante más bien se limita a ver una simulación predefinida pues no siempre se le permite interactuar con ella o su intervención consiste en decidir el momento en el que inicia la simulación, detenerla en algún punto o únicamente puede realizar

Laboratorio virtual de Física: Desarrollado por la National Taiwán Normal University [31] y cedidos para la Universidad Federal de Santa Catarina, en Brasil, el Laboratorio Virtual de Física consiste en una serie de experimentos relacionados con mecánica, dinámica, oscilaciones y ondas, termodinámica, electromagnetismo, óptica, Física Moderna, entre otros. La mayoría de los experimentos está en idioma inglés y otros en portugués. El usuario tiene poco control sobre los componentes del experimento.

El Laboratorio Virtual: Creado por la Universidad de Valladolid [44], el Laboratorio virtual es una aplicación en 2D elaborada con *applets* de Java. Presenta diversas simulaciones como por ejemplo: péndulo simple, resorte con oscilaciones amortiguadas, resorte con oscilaciones forzadas,

caída paracaidista, péndulo físico amortiguado, etcétera. En las simulaciones, el usuario puede modificar diversos parámetros como longitud, aceleración, coeficiente de amortiguación y factor de tiempo. Proporciona una serie de gráficas a través de las cuales es posible analizar el tipo de movimiento que se genera.

De acuerdo con lo anterior, es posible notar que la mayoría de los simuladores de Física disponibles, presentan los experimentos sobre un espacio 2D que, a pesar de que no todos permiten que el usuario tenga un control absoluto sobre los parámetros de cada fenómeno, sí logran mostrar una perspectiva general de su comportamiento, por otro lado, hay otros simuladores que sólo presentan animaciones prefabricadas que limitan al usuario a ser un simple observador, mientras que otros proporcionan prácticas prefabricadas modificables así como múltiples herramientas para no sólo observar y manipular el fenómeno físico, sino también permiten graficar los datos de los experimentos e incluso, fabricar videos. A pesar de las numerosas diferencias que existen entre estas herramientas, todas coinciden en el objetivo de acercar a las personas al estudio de la Física de una manera lúdica pero sin perder de vista la seriedad de los temas analizados.

En la Tabla 4 se muestra una comparación general entre los simuladores de Física anteriormente descritos y el sistema de manipulación gráfica destacando que el principal problema con los editores más poderosos como *Interactive Physics*, *Newton* y *Crocodile* es que es necesario pagar una licencia para obtenerlos, están en un idioma distinto al español, no presentan los experimentos en 3D o no permiten generar videos.

Por otro lado hay que destacar que el *sistema de manipulación gráfica* tiene importantes características que lo ponen en ventaja respecto a las otras herramientas: es en 3D, es libre, genera videos, está en español y permite el control de los parámetros de los experimentos.

Nombre Motor	Es- pa- cio	Temas	Genera video	Libre	Nivel	Idioma
Sistema de Manipulación Gráfica	3D	Mecánica Clásica	Sí	Sí	Medio superior	Español
Newton Explore Physics	3D	Cinemática y dinámica	Sí	No	Medio básico, medio superior	Inglés
Java Physics Applets	2D y 3D	Mecánica, electrodinámica, astrofísica, entre otros	No	La mayoría	Básico, medio, superior	La mayoría en inglés
Interactive Physics	2D	Mecánica, ondas, óptica, electrodinámica	No	No	Medio básico, medio superior, superior	Español y otros
Crocodile Physics	2D	Electricidad, movimiento y fuerza, óptica, ondas	No	No	Medio superior	Inglés
Phun	2D	Múltiples, principalmente mecánica	No	Sí	Básico, medio	Inglés
Laboratorio virtual de Física	2D	Física general	No	Sí	Medio superior	Portugués
Laboratorio virtual	2D	Péndulo, oscilaciones, resortes	No	Sí	Medio superior, superior	Español

Tabla 4.1: Comparación entre Simuladores de Física

4.3. El Sistema de Manipulación Gráfica de Fenómenos Físicos Reales

El aprendizaje de la Física, como ciencia experimental, exige que el alumno sea capaz de relacionar los conceptos teóricos con el comportamiento de los sucesos que observa, por lo tanto, no basta con observar los fenómenos y medirlos, es necesario además, someter un mismo fenómeno a distintas condiciones y generar las hipótesis necesarias que validen su comportamiento.

La idea de desarrollar el *Sistema de Manipulación Gráfica de Fenómenos Físicos Reales, Asistidos por Computadora* [35], tema principal de esta tesis, surge a partir de la necesidad de crear una herramienta que permita a los alumnos experimentar con un fenómeno físico a través de la manipulación de sus parámetros predeterminados y observar los cambios ocurridos en consecuencia.

El Sistema de manipulación gráfica, es una aplicación multimedia que consiste en un Simulador de eventos físicos que presenta, a través de un ambiente virtual de escritorio, una serie de experimentos relacionados con la asignatura de Mecánica Clásica. Permite a los alumnos no sólo observar sino también modificar los parámetros de cada experimento, obteniendo una respuesta gráfica en tiempo real visualizando la simulación sobre un espacio 3D, además, el sistema permite generar videos de los experimentos realizados en formato AVI, los cuales el alumno podrá conservar para visualizarlos dentro del LBV y comparar el fenómeno real, con el fenómeno simulado.

La comparación entre el modelo simulado y la realidad estimula a los alumnos a estudiar los fenómenos desde una perspectiva objetiva, porque cuando se observa un fenómeno físico, éste se analiza en un principio a partir de creencias o conocimientos previos y, una vez que se tiene acceso a la experimentación, el estudiante podrá abstraer los conceptos teóricos de Física y convertirlos en un conocimiento razonado.

El Simulador es entonces, una herramienta para experimentar, manipular propiedades físicas, observar comportamientos, generar videos y comparar con la realidad.

4.4. Características del Simulador

El objetivo del *Sistema de manipulación gráfica de fenómenos físicos reales, asistidos por computadora*, como ya se ha dicho, es cubrir los aspectos prácticos de un curso de Física a nivel bachillerato, en especial en la materia de Mecánica Clásica, y permitir a los estudiantes experimentar con simulaciones relacionadas, tal como si los experimentos se estuvieran realizando en un laboratorio. Por lo tanto, con la finalidad de alcanzar el objetivo planteado, el sistema debe tener las siguientes características:

- Debe ser un simulador interactivo.
- Debe representar experimentos relacionados con la Mecánica Clásica.
- Los experimentos deben representarse en un espacio tridimensional.
- Debe permitir a los alumnos manipular las variables físicas de los objetos que componen cada experimento.
- Debe proporcionar una respuesta gráfica en tiempo real de acuerdo a la manipulación de parámetros efectuados por el usuario.
- Finalmente, debe poseer un módulo que permita generar videos de los experimentos que hayan realizado los alumnos.

Cabe destacar que otra de las características importantes del *sistema de manipulación gráfica* es la de ser una herramienta que no pretende sustituir a la *Enciclopedia de la Física* ni al *LBV*, sino por el contrario, es una herramienta que se complementa con los otros dos materiales, tal como se observa en la Figura 4.2.

Cuando el alumno observa el video de un fenómeno físico real a través del *LBV*, comienza a generar hipótesis que expliquen el comportamiento del fenómeno que estudia a partir de sus conocimientos previos, ya sea por experiencias, por tener información teórica de la *enciclopedia*, del libro de texto, etcétera y realiza anotaciones. Cuando pasa a la etapa de experimentación dentro del *simulador*, el alumno tiene la oportunidad de estudiar el mismo fenómeno físico pero ahora será él quien tenga el control de las condiciones bajo las cuales se está desarrollando el experimento y corrobore sus hipótesis

4.5. SOFTWARE UTILIZADO EN EL DESARROLLO DEL SISTEMA 81

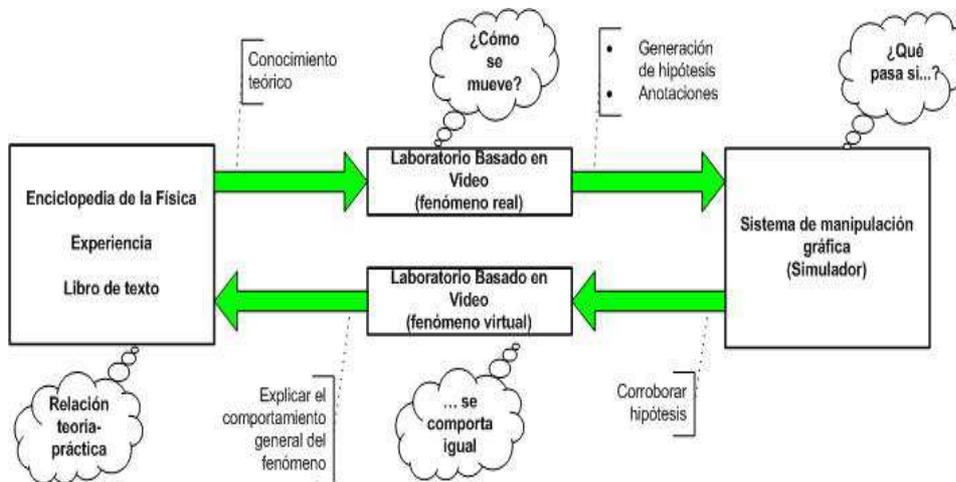


Figura 4.2: Relación entre las tres herramientas

o genere otras nuevas, finalmente, el alumno tendrá la facilidad de fabricar un video del experimento virtual y más tarde someterlo a mediciones dentro del *LBV*, para que de este modo pueda compararlo con el experimento real; por lo tanto, el alumno podrá entender y explicar el comportamiento general del fenómeno físico.

4.5. Software utilizado en el desarrollo del sistema

Quizá uno de los mayores retos a los que se enfrentó el equipo de desarrollo del Sistema de manipulación gráfica, fue precisamente la selección de las mejores herramientas de entre la gran cantidad de software disponible para fines similares. Por lo tanto, fue necesario iniciar la selección de herramientas de acuerdo a las necesidades y prioridades existentes en el momento.

Por un lado se analizaron diversos lenguajes de programación que podrían ser los adecuados para el desarrollo del sistema: VisualBasic, Java y C++. La desventaja es que la curva de aprendizaje de los mismos es larga y obliga a enfocarse en la herramienta más que en el producto, y la prioridad era crear un prototipo rápido del sistema.

A partir de entonces, se recurrió a las herramientas de autoría Authorware y Director porque precisamente estas facilitan la construcción de prototipos de sistemas interactivos de una manera rápida y eficiente. La evaluación de ambas herramientas se hizo tomando en cuenta las siguientes necesidades:

- Permitir la integración de programación.
- Proporcionar los medios necesarios para desplegar objetos tridimensionales.
- Capacidad de interactuar con herramientas que posibilitan el control de la dinámica de los objetos.
- Compatibilidad con software especializado en modelado y diseño de ambientes virtuales.

A la par de este análisis se evaluaron otras herramientas para propósitos específicos:

- 3D Studio MAX para la construcción de elementos tridimensionales.
- Shockwave 3D para desplegar el ambiente tridimensional.
- Havok Physics Engine y Reactor para controlar la dinámica de los elementos de la escena.

Finalmente, las herramientas que se describirán a continuación, resultaron ser las más eficientes para la construcción del sistema: Director y su lenguaje de programación Lingo, Shockwave3D y Havok.

4.5.1. Director

El programa Director² es una herramienta esencial en el desarrollo profesional de aplicaciones multimedia e idónea para el desarrollo del *sistema*

²La versión utilizada en este trabajo es Macromedia Director MX 2004 o Director 10, actualmente el software es propiedad de Adobe Systems Incorporated por lo que ahora se le conoce como Adobe Director MX 2004.

4.5. SOFTWARE UTILIZADO EN EL DESARROLLO DEL SISTEMA 83

de manipulación gráfica porque posee su propio lenguaje de programación orientado a objetos llamado *Lingo* a través del cual fue posible crear, para cada experimento, los objetos tridimensionales sobre un espacio *shockwave3D*, asignarles propiedades físicas invocando funciones de *Havok* y además controlarlos a nivel interfaz.

El control de las aplicaciones multimedia a través de *Director* es posible porque ofrece una serie de comportamientos (*behaviors*) que son pequeños scripts programados en Lingo y el empleo de *Xtras* programados en *C++* que pueden ser modificados para adaptarse a las necesidades de cada sistema.

Otra de las razones por las que se eligió *Director* es porque posee una de las más rápidas máquinas de *rendereo*, soporta una amplia variedad de tipos de datos multimedia como Flash, Real, Quick Time, AVI, entre otros; permite el control de cámaras; acepta modelos 3D elaborados desde Maya y 3D Studio MAX y además incluye editores básicos para texto, mapa de bits, vectores y sonido.

Las aplicaciones desarrolladas con esta herramienta pueden distribuirse en archivos ejecutables *.exe* y utilizarse en cualquier computadora sin la necesidad de tener instalado en el sistema Director o alguno de sus componentes, lo que facilita su distribución a través de cd's o cualquier otro dispositivo extraíble; por otro lado, también es posible distribuir las aplicaciones a través de Internet mediante el visor de películas de Director: *Shockwave Player*.

Más adelante, durante la explicación de la estructura del *sistema de manipulación gráfica*, se hará referencia a algunas ventanas de Director, por lo que es necesario entender cómo es la interfaz de esta herramienta. El nombre del programa es acorde a la interfaz del mismo, la idea es crear una película *Movie*, para ello existen varias ventanas: una para los actores *Cast*, otra para el montaje *Score*, otra para los guiones *Scripts* y otra para ver los resultados *Stage*. Es el desarrollador el que controla todos los aspectos de la película, convirtiéndose en el director de la misma.

4.5.2. Shockwave

Es un *plug-in* que permite visualizar las películas desarrolladas en Director, es muy popular en el desarrollo de juegos y presentaciones con fines edu-

cativos o comerciales. Contiene un motor 3D Shockwave3D y una máquina de render muy rápida, permitiendo ampliar su funcionalidad a través de los Xtras. Es de distribución gratuita y puede descargarse del sitio oficial de Shockwave [42].

4.5.3. Havok Physics Engine

En el desarrollo de este trabajo se utilizó la versión 1.0 de *Havok Physics Primer* [21], este Xtra es una extensión de Shockwave Player liberado conjuntamente con Sockwave3D. Es una herramienta de alto desempeño ideal para crear ambientes tridimensionales en donde interactúan los objetos que lo conforman; realiza los cálculos necesarios para determinar los resultados de las colisiones dependiendo de las propiedades de los objetos tales como masa, velocidad, rotación, etcétera.

Havok es ejecutado en código compilado, por lo tanto es más rápido que crear un motor de física 100 % en Lingo, además tiene la ventaja de que su código puede modificarse mediante programación en Lingo, permitiendo a los desarrolladores ajustar los parámetros de una simulación sin necesidad de programar todas las funciones completas y enfocarse tanto en las reglas del juego como en la interacción con el usuario. Es de distribución gratuita y puede descargarse del sitio *director-online* [9] junto con la documentación necesaria.

4.6. Estructura del Simulador

Para desarrollar un simulador que logre mostrar a los estudiantes la evolución de un fenómeno físico que sea controlado por ellos mismos, primero es necesario crear un ambiente sobre el cual se construirá la escena y se visualizará la simulación, también se deben proporcionar los controles que permitan manipular el estado del fenómeno y finalmente, se deben ofrecer las herramientas necesarias para poder grabar un video del experimento estudiado que podrá almacenarse para un futuro uso. Por lo tanto, el desarrollo del Simulador se ha dividido en cuatro etapas:

1. **Creación de la escena:** es la etapa en la que se construyen los objetos y se definen los parámetros predeterminados para cada experimento sobre un ambiente tridimensional donde se visualiza la evolución del sistema.
2. **Control de objetos y variables físicas:** corresponde a la etapa de manipulación de los parámetros iniciales de los objetos dinámicos, de las variables físicas y de la posición de la cámara, asegurándose de proporcionar al estudiante una respuesta gráfica en tiempo real sobre el espacio 3D.
3. **Simulación:** una vez establecidos los parámetros iniciales para un experimento, se ejecuta la simulación del fenómeno físico. Esta etapa consiste en hacer todos los cálculos necesarios que describan el comportamiento del fenómeno como consecuencia de la modificación de parámetros.
4. **Creación del video de la simulación:** esta etapa consiste en fabricar un video a partir de las imágenes generadas en cada refresco de pantalla durante la ejecución de la simulación, permitiendo mostrar una vista previa del video y guardarla en el disco duro de la computadora.

4.6.1. Creación de la escena

Cuando se precisa representar un fenómeno físico a través de una simulación, es necesario crear un modelo que logre simplificar el sistema real. En el *sistema de manipulación gráfica* se generan los experimentos a partir de un espacio tridimensional Shockwave3D, este espacio representa el mundo donde se visualizarán los objetos o componentes del sistema en estudio y un espacio en blanco de Havok, que indica que a dicho sistema se le aplicarán algunas propiedades físicas; se establecen los parámetros de iluminación y algunas variables físicas, que no siempre proporcionan una salida gráfica en tiempo real cuando se modifican pero cuya acción se verá reflejada en el espacio 3D durante la simulación.

La manera más sencilla de representar gráficamente los componentes de un sistema relacionado con la Mecánica Clásica, es a través de una serie de objetos que consisten en formas básicas o primitivas muy simples, como esferas

y cubos, ofreciendo al alumno la posibilidad de trabajar con experimentos que no requieren demasiadas modificaciones.

Un objeto tridimensional hace uso de geometrías 3D que son visibles únicamente cuando son utilizadas por un modelo que ocupa una posición y orientación dentro de un ambiente 3D, dichas geometrías reciben el nombre de modelo fuente. Por lo tanto, los objetos tridimensionales se generan haciendo referencia al modelo fuente a través de la propiedad *modelResource*. En Director existen una serie de modelos predefinidos o primitivas, como esferas, cubos, cilindros y sistemas de partículas.

La creación de la escena comienza con la declaración del ambiente 3D y la generación de modelos tridimensionales que se volverán visibles en el mismo de la siguiente manera:

ambiente3D.newModelResource(“nombreModelo”, #tipo) donde el atributo *#tipo* se refiere al tipo de primitiva que se empleará.

Una vez creados los objetos tridimensionales es posible asignarles propiedades físicas; cada uno de ellos es manejado como un cuerpo rígido, es decir, que nunca se deforma. Esta propiedad facilita que el alumno se concentre en el fenómeno y no en la posible deformación de los objetos como reacción a las fuerzas aplicadas sobre ellos a lo largo del tiempo.

Los cuerpos rígidos son instancias de Havok que se crean a partir del modelo desplegado en el ambiente 3D cuya forma depende del tipo de primitiva que se haya seleccionado. Para que Havok pueda acceder a la información de la geometría del objeto, es necesario agregar un modificador de forma llamado *meshdeform*; función que estipula que la primitiva utilizada, ya sea una esfera o un cubo, es la misma que deberá usar en la simulación:

modelo.addModifier(#meshDeform)

De acuerdo a su función, los cuerpos rígidos pueden ser de dos tipos: móviles y fijos, ver Figura 4.3.

Por un lado, los *cuerpos rígidos móviles* son aquellos objetos que reaccionan ante las fuerzas físicas tales como la gravedad, la fricción, etcétera; en consecuencia, su posición y orientación cambiarán porque es posible que estos colisionen ya sea con un objeto rígido fijo o bien, con otro objeto rígido

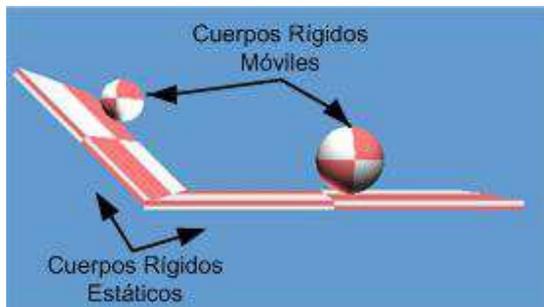


Figura 4.3: Ambiente 3D sobre un espacio Shockwave3D

móvil durante la simulación. Se definen de la siguiente manera:

`havok.makeMovableRigidBody(crNombre, masa, isConvex, #tipo)`

donde:

crNombre: es el nombre del objeto o cuerpo rígido;

masa: se refiere a la cantidad de materia en un objeto y su resistencia al cambio en movimiento. Cuando se establece que un objeto es móvil, es necesario definir su masa cuyo valor predeterminado es de 1 kilogramo;

isConvex: es una bandera booleana (*true/false*) que indica si el objeto rígido es cóncavo o convexo; en este caso, los objetos son considerados como convexos por defecto;

#tipo: se refiere al tipo de primitiva del objeto, comúnmente se utilizan esferas y cubos. Por defecto, en los experimentos los cuerpos rígidos móviles son del tipo esfera (*#sphere*).

Por otro lado los *cuerpos rígidos fijos*, a diferencia de los móviles, no reaccionan ante fuerzas físicas como la gravedad o ante la colisión contra otro objeto sino que siempre permanecen en la misma posición durante la simulación. Su definición es similar a la de los cuerpos rígidos móviles salvo que no es necesario definir el parámetro masa. Por defecto en el sistema de manipulación gráfica, los cuerpos rígidos fijos son del tipo caja (*#box*):

havok.makeFixedRigidBody(crNombre, isConvex, #tipo)

La visibilidad de los objetos de la escena depende del manejo de los modelos de iluminación y de las cámaras. En el sistema de manipulación gráfica se ha definido una luz de tipo ambiental (*#ambient*) que se aplica sobre los modelos y otra direccional que se aplica a aquellas partes sobre las que apunta la luz, está situada en la parte superior central (*#topCenter*) del ambiente 3D por lo que ofrece una iluminación por encima de los objetos haciendo que su distancia a los mismos, sea irrelevante:

newLight(nombreLuz, #tipo): se crea una luz

member(“nombreLuz”).directionalpreset = #topCenter: se establece la dirección en que debe apuntar la luz.

Las cámaras son instancias asociadas al ambiente 3D que proporcionan diferentes vistas del mismo siendo posible definir más de una a la vez, en este caso, se ha definido una cámara cuya posición está a una distancia de 30 unidades sobre el eje Y y a 250 unidades sobre el eje Z, estos parámetros podrán modificarse por el usuario:

member(“ambiente3D”).cameraPosition = vector(0,0, 30,0, 250,0)

Finalmente, se definen las variables físicas que afectarán tanto al ambiente virtual como a los objetos, estas se miden en metros, segundos y kilogramos pero también es posible medirlas en pulgadas y kilómetros, ver Tabla 5. Estas variables se dividen de acuerdo al control que ofrecen en dos categorías: control global y control local. Las variables físicas de control global son aquellas que afectan a la escena en general, por ejemplo la gravedad; las variables de control local son aquellas que afectan únicamente a los cuerpos rígidos sobre las que se aplican, por ejemplo masa, volumen, fricción, restitución, entre otras.

Variable física	Valor predefinido	Definición
Gravedad: factor de aceleración de caída de los cuerpos rígidos por la atracción gravitacional.	$-9,81 \frac{m}{s^2}$	Havok.gravity=vector(0.00,-9.81,0.00)
Masa: es la cantidad de materia de los cuerpos en movimiento (aplicable a cuerpos rígidos móviles)	$1kg$	havok.rigidBody(crNombre).mass = 1
Volumen: establece el tamaño de las masas -esferas- (aplicable a cuerpos rígidos móviles)	<i>Radio = 10cm</i>	ambiente3D.modelResource(crNombre). ...radius = 10
Fricción: fuerza requerida para mover o rodar un cuerpo rígido sobre otro (aplicable a cuerpos rígidos móviles)	Oscila entre 0 y 1	havok.rigidBody.friction = 0.3
Restitución: aumento o disminución de energía después de cada colisión (aplicable a cuerpos rígidos móviles)	0 = no rebota, 0.5 = perder el 50% de energía después de cada salto, 1 = rebote perfecto, por lo tanto, oscila entre 0 y 1	havok.rigidBody.restitution = 1

Tabla 4.2: Definición y valores de las variables físicas

4.6.2. Control de objetos y variables físicas

Cuando el estudiante entra a un experimento determinado, observa la escena bajo una serie de parámetros predefinidos, por lo tanto, en esta etapa, es necesario proporcionarle la posibilidad de ajustar dichos parámetros de acuerdo a su preferencia. Al igual que las variables físicas, se hará una distinción entre los parámetros que afectan la escena en general y los que afectan únicamente a los objetos.

Por un lado, los parámetros que afectan a la escena en general son la posición de la cámara y la variable de gravedad. La posición de la cámara se modifica mediante una transformación de posición en un incremento de una unidad sobre los ejes X y Y (desplazamiento vertical y horizontal) o bien transformando su posición sobre el eje Z que da un efecto de acercamiento (zoom in, zoom out), ver Tabla 6.

Movimiento de cámara	de	Modificación vectorial	
Desplazamiento vertical		<i>ambiente3D.camera.translate</i> <i>vector(0,0, ±1,0, 0,0)</i>	=
Desplazamiento horizontal		<i>ambiente3D.camera.translate</i> <i>vector(±1,0, 0,0, 0,0)</i>	=
Desplazamiento <i>zoom in, zoom out</i>		<i>ambiente3D.camera.translate</i> <i>vector(0,0, 0,0, ±1,0)</i>	=

Tabla 4.3: Movimiento de cámara

Por otro lado, el valor de la gravedad que será aplicada sobre el sistema se maneja como un vector cuyo valor varía en sentido negativo sobre el eje Y de acuerdo a los valores gravitatorios de los planetas del Sistema Solar incluyendo al Sol y a la Luna; esta función no produce una respuesta gráfica en tiempo real puesto que, contrario a lo que sucede con el manejo de cámaras, será visible hasta el momento en que se esté ejecutando la simulación:

$$\text{havok.gravity} = \text{vector}(\mathbf{0}, \mathbf{0}, -\text{gravAstro}, \mathbf{0}, \mathbf{0})$$

Respecto al control de los parámetros de los objetos, el estudiante podrá manipular algunas propiedades físicas de los objetos móviles de los experimentos, entre los que se pueden enumerar a la posición de las masas en el

espacio, su volumen, su masa y en algunos casos, podrá modificar el ángulo de inclinación de los objetos fijos.

En todos los experimentos es posible variar la posición de las masas de acuerdo a su posición (x, y) en el espacio 3D, incrementando este valor en una unidad ya sea en el eje X o en el eje Y de acuerdo las necesidades del estudiante:

ambiente3D.model(crNombre).transform.position = vector(WorldPosition.x, WorldPosition.y, WorldPosition.z), donde:

WorldPosition: se refiere a la posición relativa del objeto dentro del espacio 3D.

La masa de los objetos está determinada por la propiedad *mass* descrita anteriormente ; éste valor se incrementa en una unidad y siempre deberá ser mayor que $0Kg$. porque de .lo contrario, el sistema asumirá que se trata de un cuerpo rígido fijo:

ambiente3D.rigidBody(crNombre).mass

El volumen de las masas está determinado por el valor del radio utilizado para generar el modelo fuente, es decir, el radio de una esfera, cuyo valor por defecto es $radius = 10$ pero podrá transformarse incrementando o disminuyendo en una unidad este valor siempre y cuando sea mayor que 0:

ambiente3D.modelResource(crNombre).radius

Las variables físicas de fricción y restitución son parámetros que no podrán ser manipulados por el usuario por lo que sus valores permanecerán igual en todas las simulaciones, es decir, *fricción* = 0,3 y *restitución* = 1, este último significa que las masas tendrán un rebote perfecto.

En el caso de los cuerpos rígidos fijos, el usuario podrá variar el ángulo de inclinación de las rampas (en los experimentos de colisión de masas, movimiento rectilíneo uniforme y movimiento uniformemente acelerado), la transformación se hace en el eje Z respecto a su posición relativa en el espacio 3D:

```
ambiente3D.model(crNombre).transform.rotation =  
  
vector(0,0, 0,0, worldPosition.z)
```

En general la manipulación de parámetros iniciales de una simulación responde a una serie de comportamientos que disparan acciones a ejecutarse mediante el envío de señales, por ejemplo, si se presiona el botón “Venus”, se envía un mensaje a la función correspondiente para indicar que el valor de la gravedad del sistema será de acuerdo al planeta Venus; el mismo comportamiento se aplica a cada uno de los parámetros que el estudiante podrá modificar.

4.6.3. Simulación

La etapa de simulación es la más importante en el *sistema de manipulación gráfica*, es en este momento donde el fenómeno físico se verá afectado por los parámetros definidos por el alumno, se aplicarán las fuerzas físicas, se detectarán y resolverán las colisiones efectuadas actualizando el despliegue a pantalla cada determinado tiempo.

El proceso de simulación se lleva a cabo mediante el *motor de física Havok*, el cual es necesario inicializar para que sea posible asignarles propiedades físicas a los objetos:

havok.initialize(ambiente3D, tolerancia, escala) donde:

ambiente3D: indica el nombre del espacio Shockwave3D sobre el cual se harán cálculos con Havok;

tolerancia: éste parámetro, llamado *tolerancia de colisión*, (ver Figura 4.4) define la distancia a la que pueden estar dos objetos sin tocarse;

escala: define las unidades en las que el mundo se está midiendo.

El parámetro *Tolerancia de colisión* se refiere al valor de la distancia mínima a la que pueden estar dos objetos sin tocarse durante la simulación.

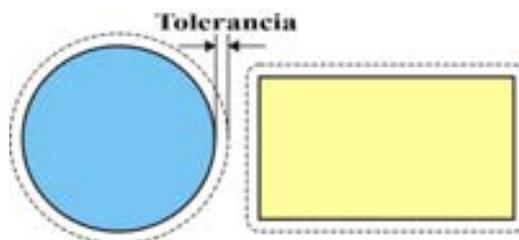


Figura 4.4: Tolerancia de colisión

Este parámetro se establece con la finalidad de evitar interpenetraciones dado que la detección de colisiones se hace después de que los parámetros iniciales se han modificado y cuando un objeto penetra en otro, la colisión es ignorada y el objeto simplemente lo traspasa; este factor está medido en las mismas unidades de la escala utilizada en la escena, por lo tanto como regla general, es preferible fijar este valor cerca del 10% del factor de escala utilizado para medir el mundo en la simulación, en este caso, la escala está en metros y equivale a 1.0 en el sistema, por lo que la tolerancia de colisión es 0.1.

Dado que los experimentos presentados en el *sistema de manipulación gráfica* son simulaciones que se encuentran en un estado de evolución continua, es decir, los objetos se están moviendo, colisionando y reaccionando todo el tiempo, entonces es necesario mapear el comportamiento del sistema en una serie de fotos instantáneas para generar con ellas una animación que exhiba dicha evolución; en este caso la animación es generada a 30 cuadros por segundo, esto significa que el motor de física muestra en pantalla la evolución del mundo cada $\frac{1}{30}$ de segundo.

Havok es una herramienta que permite hacer estimaciones referentes al nuevo estado de los objetos de acuerdo con las fuerzas que actúan sobre estos. El valor del estado inicial de cada objeto es conocido porque es el predefinido por los estudiantes, pero además se conocen los valores de las fuerzas que están actuando sobre ellos, por otro lado, hay que tomar en cuenta que tales fuerzas raramente son constantes, por ejemplo, el valor de la gravedad se mantiene constante casi todo el tiempo, pero la mayoría de las otras fuerzas como el viento o la resistencia del aire, no lo son. Por lo tanto para hacer estas predicciones sin perder mayor detalle de lo que pueda ocurrir con las fuerzas que no son constantes y evitar descartar variaciones importantes en la posición de los objetos, es necesario dividir la simulación en pasos de tiempo

time steps [21]. Cada paso de tiempo es un periodo en el cuál se harán las estimaciones entre la posición (t_n) y la posición (t_{n+1}) de cada objeto, dicho periodo está dado en h segundos y su tamaño afectará la precisión de la simulación.

En general, mientras más pequeño sea el valor del *time step* para estimar las posiciones de los objetos, mayor será la precisión del resultado al final de este periodo, pero si se desea dar varios pasos adelante en el tiempo (t) es mejor dividirlo entre n pasos de un intervalo más pequeño $\frac{t}{n}$.

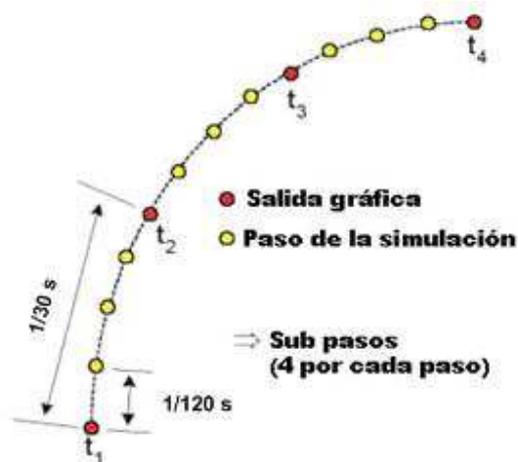


Figura 4.5: Pasos y subpasos en una simulación

La idea es tomar *pasos de tiempo* pequeños, evaluar todas las fuerzas que actúan sobre los objetos, determinar las nuevas posiciones, velocidades y otros parámetros de los objetos al final de dicho periodo y comenzar de nuevo. Finalmente lo que se obtiene es una serie de fotos instantáneas que muestra cómo ha evolucionado el estado del sistema. Dado que la película se está reproduciendo a 30 cuadros por segundo, es decir, la pantalla se refresca a 30 Hz, el valor del *paso de tiempo* es $\frac{1}{30}$, este valor ofrece una buena precisión, pero para lograr mayor exactitud, es posible dividir este valor sin necesidad de refrescar la pantalla mediante sub pasos *substeps*.

El parámetro de los *sub pasos* especifica el número de pasos que el motor toma antes de actualizar el espacio, ofreciendo control sobre la simulación sin necesidad de refrescar la pantalla y obtener demasiadas imágenes que

posiblemente no son necesarias, ver Figura 4.5. Si $substeps = 0$ entonces no se toman pasos físicos, con $substeps = 1$ se toma un único paso de simulación para cada actualización a pantalla, en el caso de este sistema, se tomarán cuatro pasos físicos antes de actualizar la pantalla, es decir $substeps = 5$.

El parámetro de pasos y subpasos se define una vez que Havok ha sido inicializado:

havok.step(incrementoTiempo, numeroSubpasos)

donde:

incrementoTiempo: se refiere al periodo en el que se dividirá la simulación y

numeroSubpasos: se refiere a que ese incremento de tiempo deberá dividirse en unidades más pequeñas, por ejemplo, si la película se reproducirá a 30cps, entonces el incremento de tiempo en la simulación será de 0.03 segundos por cada cuadro $\frac{1,0}{30}$.

Cuando el usuario ha definido los parámetros iniciales para el experimento y presiona el botón *simular*, Havok se encarga de desempeñar lo que se conoce como el ciclo de una simulación del cual se habló en el Capítulo 2 y que se inicia una vez que el usuario ha definido los parámetros iniciales para cada experimento. De acuerdo con la Figura 4 6, el ciclo de la simulación [21] consiste en lo siguiente: detectar colisiones, actualizar fuerzas, integración de datos para definir las nuevas posiciones de los objetos, desplegar en pantalla los cambios y recorrer el tiempo.

- a) **Detección de colisiones:** En una simulación física es necesario determinar, si es el caso, cuáles objetos han chocado porque esto resultará en nuevas fuerzas de colisión y fricción que serán introducidas al sistema.
- b) **Actualización de fuerzas:** Actualiza todas las fuerzas que actúan sobre cada uno de los objetos como resultado de la detección de colisiones o bien, como consecuencia de la entrada de datos por el usuario.

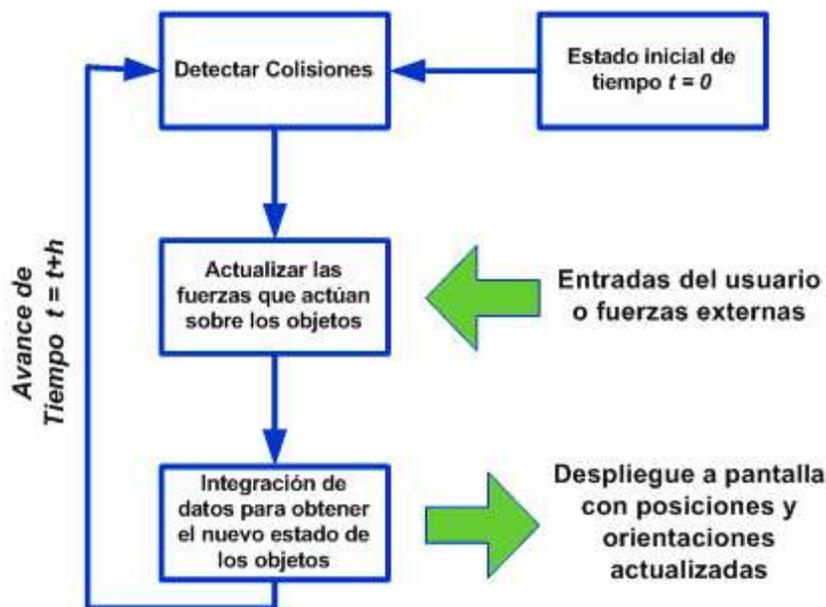


Figura 4.6: Ciclo de una Simulación

- c) **Determinación del nuevo estado de los objetos:** Habiendo acumulado todas las fuerzas, determina el nuevo estado de los objetos (posición, orientación, velocidad, aceleración) utilizando los algoritmos necesarios para realizar los cálculos matemáticos en ecuaciones diferenciales que describen el movimiento de los objetos y estimar el valor de todas las variables involucradas; dicha información es utilizada para actualizar el ambiente.
- d) **Avance de tiempo:** Una simulación es monitoreada por el motor de física paso a paso, es decir, a cada determinado intervalo, debe analizarse el estado de los objetos de la escena, entonces el parámetro de avance de tiempo está delimitado por el tamaño de h donde “ h ” representa el tamaño del paso de tiempo (*timeStep*) y determina si esas nuevas posiciones de los objetos resultan en colisiones entre cualquiera de ellos.

A medida que una simulación se vuelve más compleja, las matemáticas requeridas para calcular las nuevas posiciones y velocidades de los objetos se vuelven igualmente más complejas, corriendo el riesgo de presentar resultados

poco precisos. En Havok está implementado un rápido *integrador numérico* de series de ecuaciones diferenciales que describen el movimiento de los objetos. Un *integrador* es un algoritmo que intenta estimar el nuevo estado de una variable o parámetro, por ejemplo, la posición, conociendo información como la razón de cambio del parámetro, es decir, su velocidad.

La actividad crucial de un motor de física es la detección de colisiones, que consiste en verificar si un objeto ha chocado con otro formando pares de colisiones. Cuando se tienen n objetos en la escena, es necesario garantizar que cada par de objetos ha sido verificado, esto requiere $\frac{n(n-1)}{2}$ pruebas, es decir, para cada objeto n se hace una prueba con cualquier otro objeto menos consigo mismo, $(n - 1)$ dando $n(n - 1)$ pruebas, pero dado que una prueba en la que el cuerpo A choca con el cuerpo B es lo mismo que si B chocara con A , entonces se divide entre 2. Así que para sistemas con 4 objetos es necesario hacer 6 pruebas, para 100 objetos deberán hacerse 4950 pruebas, convirtiéndose en un proceso muy caro, es por eso que en este sistema los experimentos contienen a lo más 5 objetos con un máximo de 2 objetos móviles, reduciendo el número de pruebas a un máximo de 7 logrando buenos resultados.

Estas pruebas de colisión se hacen mediante la función *registerInterest* que especifica las colisiones entre cuerpos rígidos y envía los detalles en forma de lista para que estos sean actualizados durante el ciclo de simulación:

havok.registerInterest(crNombre1, crNombre2, frecuencia, umbral) donde:

crNombre1 y crNombre2: son los nombres de los pares de colisión;

frecuencia: determina qué tan a menudo es registrada una colisión y

umbral: especifica qué tan fuerte fue una colisión antes de invocar de nuevo la detección de colisiones para saber si el resultado de esa colisión provocó nuevas colisiones, el umbral está definido en términos de la velocidad relativa normal (en metros por segundo) de los objetos en colisión.

La lista que esta función almacena antes de volver a llamar a la función de colisión es la siguiente (ver Figura 4.7):

```
havok.collisionList(crNombre1, crNombre2, contactPoint, ...)
(... contactNormal, velocidadrelNormal)
```

donde:

crNombre1 y **crNombre2**: son los nombres de los pares de colisión;

contactPoint: es el punto de colisión entre los dos objetos;

contactNormal: es la dirección de colisión del segundo objeto en el punto de colisión y la velocidad normal relativa que es la velocidad relativa de los dos objetos involucrados en la colisión. Este valor es la suma del valor absoluto de las velocidades de los objetos en la dirección de la normal de colisión. Así que para dos esferas, cada una se mueve a $10 \frac{m}{s}$ hacia la otra, por lo tanto, la velocidad normal relativa es de $20 \frac{m}{s}$.

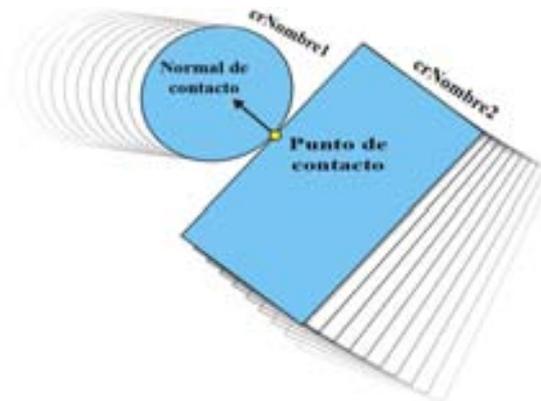


Figura 4.7: Registro de colisión

En el *sistema de manipulación gráfica* las simulaciones se visualizan precisamente como si fueran videos o animaciones, pero existe un caso particular que consiste en una función de seguimiento (ver Figura 4.8) que muestra la trayectoria de los objetos móviles de la escena generando una superposición de los pasos de salida gráfica generados por el simulador:

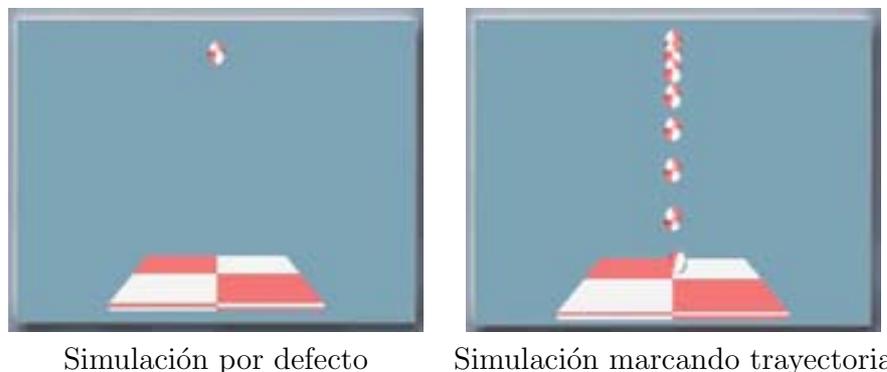


Figura 4.8: Modos de despliegue de simulación

ambiente3D.camera.colorBuffer.clearAtRender = 0

, esta función indica que al actualizarse el sistema no se borra la imagen de la posición inmediata anterior del objeto, permitiendo al usuario observar el proceso de aceleración de manera continua, permitiéndole comparar visualmente las diferentes posiciones de los objetos basándose en los rastros dejados por los pasos de la simulación, a esta función se le ha llamado *marcar recorrido* y el estudiante podrá activarla en el momento en que esté manipulando los parámetros de la escena.

Finalmente, cuando ha terminado de ejecutarse la simulación, es necesario desactivar y liberar la memoria mediante la siguiente función:

havok.shutdown()

En resumen, como se explica en la Figura 4.9, el control de una simulación en el *sistema de manipulación gráfica* comienza a partir de la inicialización de Havok, la cual detectará las geometrías de los objetos, las variables físicas y sus respectivos valores iniciales, ya sea los establecidos por el alumno o los predefinidos en el sistema, también se detecta el valor del incremento del tiempo de simulación para establecer el refresco de pantalla y los cálculos correspondientes a las posiciones de los objetos. El siguiente paso es mostrar a pantalla la evolución del sistema, detectar las posibles colisiones y resolverlas calculando fuerzas, velocidades, aceleración, etcétera, el sistema genera una lista de colisión que contendrá la información necesaria para determinar si el

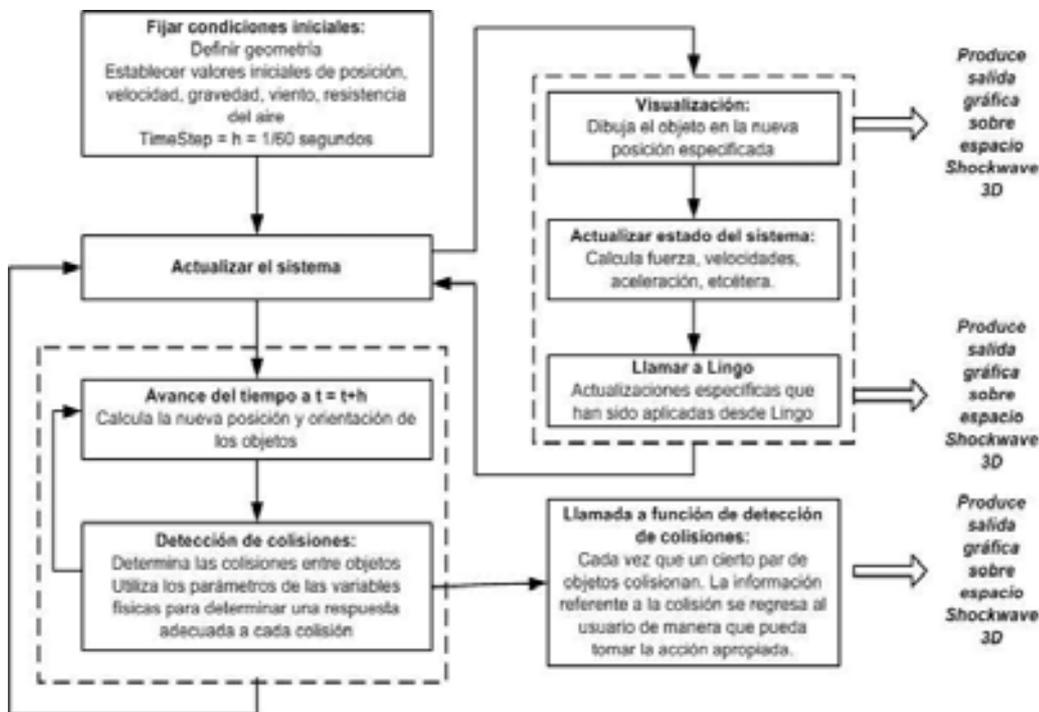


Figura 4.9: Proceso de control de una simulación

nuevo estado de los objetos ha provocado una nueva colisión y así repetir el ciclo.

4.6.4. Creación de video

Cuando se genera una simulación, en realidad se están mostrando imágenes consecutivas de la evolución del sistema, en este caso, para poder crear un video de alguno de los experimentos propuestos, basta con guardar las imágenes generadas en algún sitio para que después estas sean “pegadas” y con ello obtener un video en formato AVI que podrá almacenarse para su uso posterior.

El proceso de creación del video consiste en lo siguiente: mientras se está ejecutando el proceso de simulación, se genera una secuencia de imágenes del experimento de acuerdo con el valor del parámetro *time step*; en el caso

de este sistema dicho valor es igual a $\frac{1}{30}$ segundos, es decir, se generarán 30 imágenes por cada segundo transcurrido. Estas imágenes se almacenan tanto en la ventana *Cast de Director* como en una carpeta llamada *temp\imagen* que se generará en la unidad *C* del disco duro de la computadora del estudiante.

Las imágenes almacenadas en la ventana *Cast* son mapas de bits utilizados para generar una vista previa del video:

member(#bitmap).image=member(ambiente3D).image, esta función genera una imagen en mapa de bits del ambiente 3D, en otras palabras, como si se tomara una foto de la escena.

En la Figura 4.104 10 se muestra que las imágenes se almacenan en orden consecutivo en la ventana *Cast*, entonces es posible hacer que estas imágenes se vayan leyendo una por una para generar una animación que exhiba la evolución del sistema mediante la siguiente función:

leeImagen = member(numImagen + 1, nombreCast)

donde:

numImagen: se refiere al número de la posición que ocupa la imagen seleccionada.

nombreCast: se refiere al nombre que recibe la ventana *Cast* donde se encuentran las imágenes.

Esta lectura de imágenes genera una vista previa del video a generar con la finalidad de comprobar si el alumno desea conservarlo o desecharlo.

A la par que se generan las imágenes dentro de la ventana *Cast*, se crean también las imágenes en formato JPG dentro del disco duro de la computadora (ver Figura 4.11 4 11) mediante un xtra llamado *SharpExport* [41] que permite hacer referencia a un mapa de bits y utilizarlo como una imagen JPG:

xtra.exportJPG(member“mapaBits”, “ruta + etiqueta.jpg”, calidad)

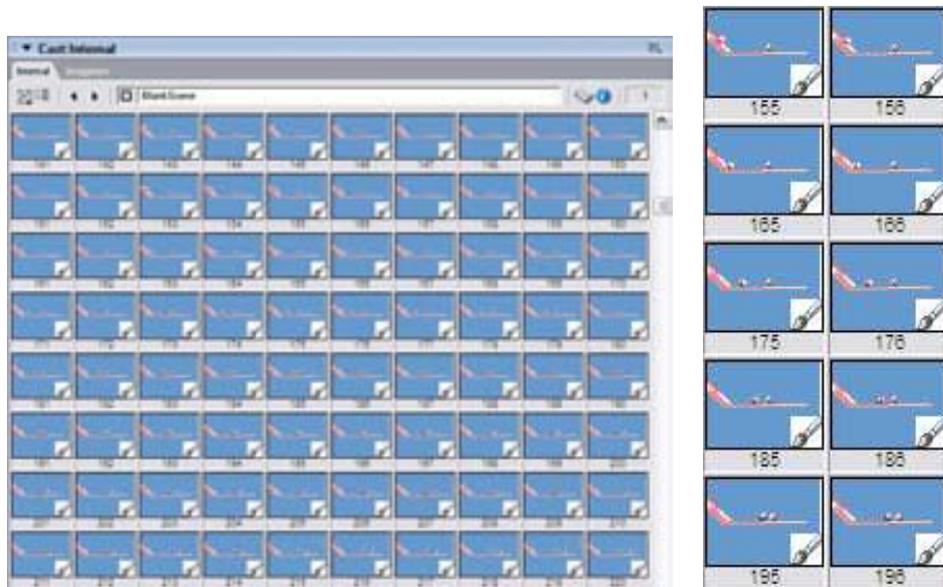


Figura 4.10: Imágenes almacenadas en la ventana Cast de Director

donde:

member “mapaBits”: es a referencia al mapa de bits almacenado en la ventana Cast;

el parámetro “ruta+etiqueta.jpg”: se refiere a la ruta donde deberán guardarse las imágenes y el nombre de las mismas seguidas del sufijo *.jpg*;

calidad: se refiere al nivel de compresión de las imágenes, mientras más grande sea este valor, mayor será la calidad.

Las imágenes se guardan en una carpeta llamada *temp\imagen* en la unidad *C*; para generar dicha carpeta se hace uso de un xtra llamado *FileXtra4* [11], dicho xtra se encarga de corroborar la existencia de la carpeta contenedora de imágenes en el disco duro y, en caso de no existir, la genera con las sentencias **folderExist** y **folderCreate** respectivamente.

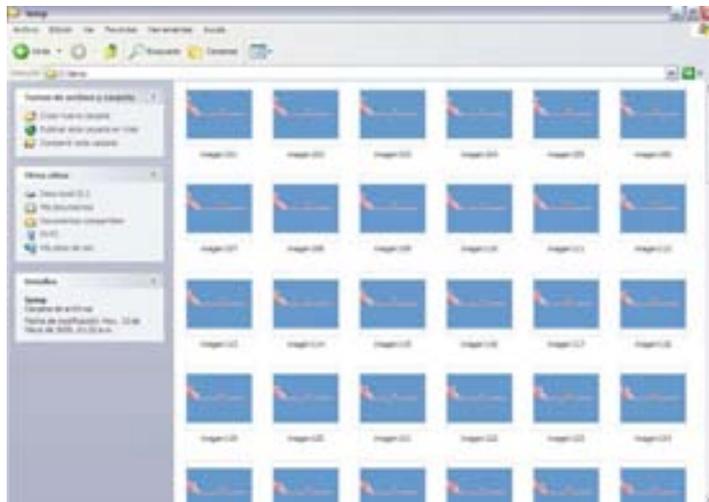


Figura 4.11: Imágenes almacenadas en la unidad C del disco duro de la computadora

Cada una de las imágenes que han sido almacenadas dentro del disco duro posee un nombre propio que corresponde a una secuencia numérica ascendente, entonces es necesario “pegarlas” para producir un video en formato AVI. Para tal fin se utiliza un *Constructor de Películas* (Figura 4.12 4 12), que es un software programado en VisualBasic donde se selecciona la primera imagen generada de la simulación, a partir de esta recorre cada una de las imágenes pegándolas ordenadamente de acuerdo a su posición en la numeración, de este modo se construye una secuencia de video que correrá a 15 o más cuadros por segundo, el cual se almacenará en cualquier carpeta que el alumno haya seleccionado. Cuando se presiona el botón *Crear AVI*, aparece una pantalla en donde se debe seleccionar el tipo de compresor con el que se fabricará el video (Figura 4.134 13) y se obtiene un video que podrá utilizarse en el LBV.

Cuando se termina de elaborar el video y se reinicia el experimento del simulador, las imágenes almacenadas tanto en el disco duro como en el Cast se eliminan: `fileRecycle(ruta+etiqueta.jpg)` utilizando el xtra FileXtra4.



Figura 4.12: Constructor de películas AVI



Figura 4.13: Compresor de video

4.7. Requerimientos mínimos del sistema

Durante el desarrollo de este proyecto se utilizó una computadora con doble procesador *Intel® Pentium® III* a una velocidad de 800 MHz, 512MB en memoria RAM, un disco duro con capacidad de 30GB y el sistema operativo *Windows® XP*, pero como resultado de varias pruebas realizadas, es posible ejecutar el programa en una computadora que cumpla con los siguientes requerimientos mínimos:

- Procesador *Intel® Pentium® II*.
- 256MB en memoria RAM.
- *Windows® 98* o superior.
- Pantalla a una resolución de 800x600 pixeles a 256 colores.

Los requerimientos anteriores son los mínimos óptimos para ejecutar satisfactoriamente la aplicación, pero si se cuenta con una computadora de mayor capacidad, entonces el desempeño del sistema mejorará notablemente.

4.8. Interacción del usuario con el sistema

Se ha considerado que el *sistema de manipulación gráfica* sea una herramienta que los estudiantes puedan utilizar en casa sin la necesidad de un profesor que les indique cómo utilizarla, por lo tanto, la interfaz se ha construido de manera tal que los alumnos puedan entender y realizar los experimentos propuestos a través de una serie de pasos sencillos ilustrados en el diagrama de la Figura 4 14.

1. Seleccionar práctica.
2. Manipular los parámetros del experimento.
3. Iniciar la ejecución de la simulación.
4. Detener la simulación.
5. Asignar nombre y ubicación al video (si es el caso).

El sistema se ha desarrollado con una interfaz que evitará que los alumnos cometan errores de uso de la herramienta navegando en a lo más tres pantallas, cada una de ellas con un propósito específico: la primera es la pantalla de *Menú principal* donde los alumnos seleccionarán un experimento; la segunda pantalla del experimento donde los alumnos permanecerán la mayor parte del tiempo pues en ella podrán modificar los parámetros del experimento seleccionado y ejecutar simulaciones; y la tercera, a la cual no siempre se accede, es la pantalla donde se muestra una vista previa del video, en caso que el estudiante haya decidido fabricar uno.

Al inicio del programa el sistema muestra el *Menú Principal* que contiene los nombres de las diferentes prácticas a las que los alumnos podrán acceder: Caída Libre, Colisión de Masas, Movimiento Rectilíneo Uniforme, Movimiento Uniformemente Acelerado, Resorte-Masa y Péndulo Simple.

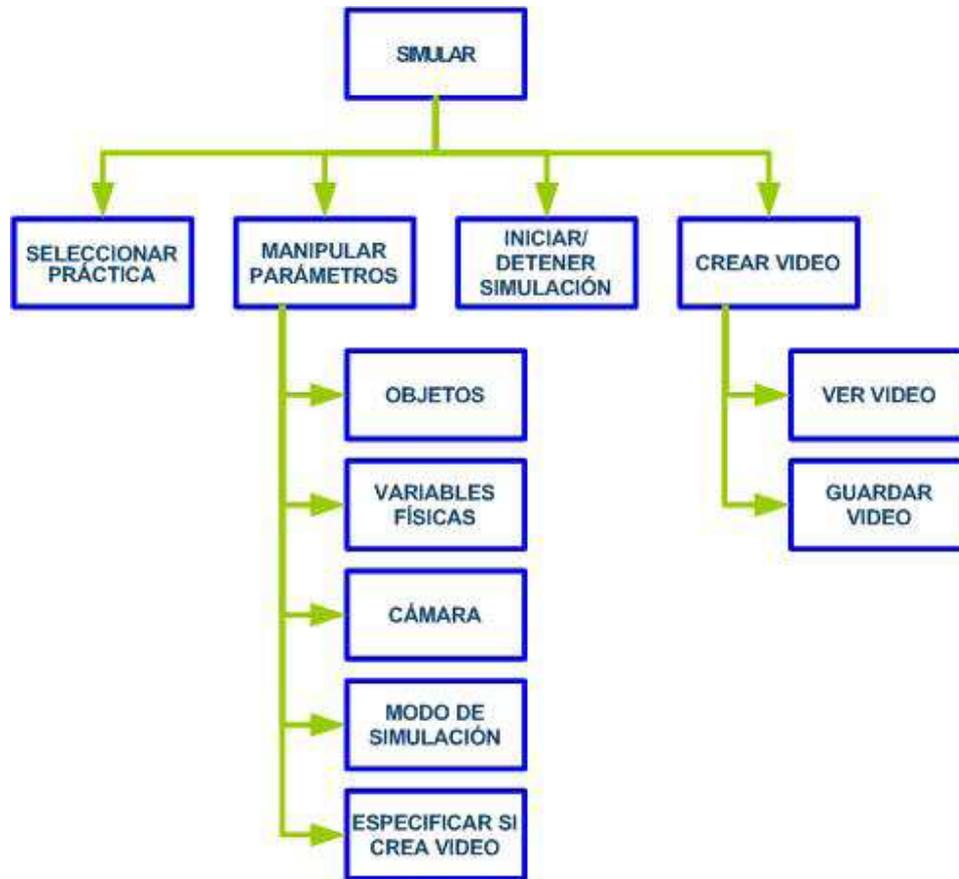


Figura 4.14: Diagrama de tareas del usuario

Una vez seleccionado el experimento, el sistema genera el espacio de la práctica correspondiente con sus respectivos controles. Es en esta sección donde el usuario podrá realizar las siguientes acciones:

1. Manipular posición de cámaras.
2. Asignar los parámetros de inicio del experimento.
3. Decidir el modo de despliegue de la simulación (opcional).
4. Decidir si se generará un video del experimento (opcional).

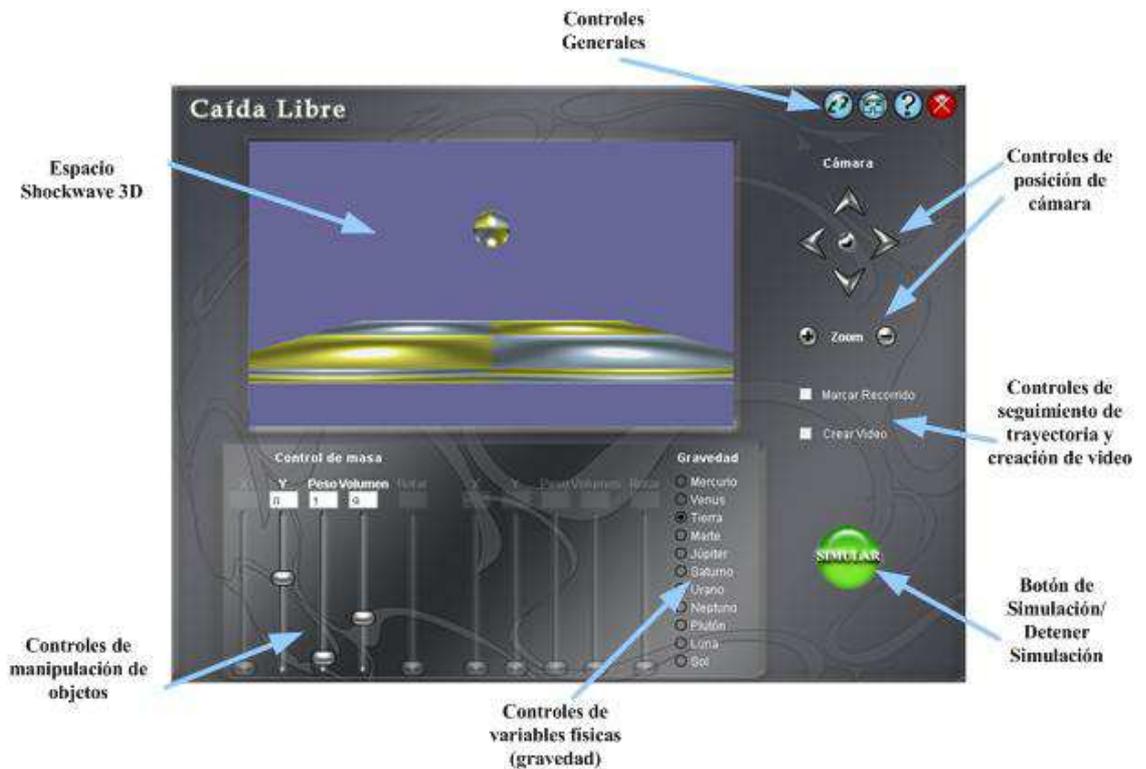


Figura 4.15: Distribución de controles del simulador

En el *sistema de manipulación gráfica*, los controles se encuentran agrupados de acuerdo a su funcionalidad, la Figura 4 15 muestra la forma general de la distribución de controles en los experimentos.

En la parte superior derecha se encuentran los controles generales del sistema, enumerados de izquierda a derecha son:

- Reestablecer: regresa los parámetros del experimento a sus valores originales al igual que regresa el ambiente 3D a su estado original.
- Home: regresa al Menú principal donde los alumnos podrás seleccionar un experimento diferente.
- Ayuda: presenta una ayuda que resume el modo en que el usuario debe

usar la herramienta.

- Salir: cierra la aplicación.

En la parte central de la herramienta se encuentra el espacio 3D sobre el cual se visualiza la escena y por lo tanto, la simulación, es en este espacio donde el usuario recibe una respuesta gráfica en tiempo real ante la manipulación de parámetros tales como posición y volumen de los objetos, así como corroborar la correcta posición de la cámara.

A la derecha del espacio 3D se encuentran los controles de la cámara que el usuario podrá posicionar de acuerdo a sus necesidades o para observar mejor algún detalle particular, pudiendo regresarla a su posición original.

En la parte inferior están los controles de los parámetros de los objetos del experimento, aquí se podrán variar la posición, la masa y el volumen de los objetos móviles, se controlará también la inclinación de las rampas en caso de ser necesario y finalmente, podrá variarse el valor de la gravedad de acuerdo a los planetas del Sistema Solar, al Sol y a la Luna, ocupando por defecto el valor de la gravedad respecto a la Tierra.

La opción *marcar recorrido* sirve para señalar la trayectoria de los objetos mientras se está ejecutando la simulación; por otro lado, la opción crear video como su nombre lo indica, sirve para generar un video del experimento que en ese momento se esté manipulando.

Una vez determinados los parámetros del experimento, el estudiante deberá presionar el botón *simulación* para arrancar en ese momento la ejecución de la simulación del evento físico, éste botón cambia de estado y se convierte en el botón *detener simulación* indicándole de este modo al sistema que debe detener la ejecución de la simulación.

Cuando el sistema no debe generar un video del experimento, entonces al detener la simulación, se restaurarán los valores predeterminados del experimento; el alumno decidirá si seguirá trabajando con ese mismo experimento o podrá regresar al *menú principal* y seleccionar otro distinto. En caso de que el alumno haya seleccionado generar un video, el sistema pasará a la ventana donde se mostrará una vista previa del video, si el video ha cumplido sus expectativas, entonces podrá guardarlo en el disco duro y asignarle un nombre. Enseguida, el sistema restaurará los parámetros del experimento y el ciclo

vuelve a empezar.

La idea de fabricar videos en la herramienta tiene el objetivo de proporcionar a los alumnos la posibilidad de comparar los resultados de los videos de las simulaciones con los del sistema real. Por un lado, el alumno podrá comprender el comportamiento general del fenómeno que estudia y por otra parte, se valida el modelo simulado de tal manera que los resultados obtenidos coinciden con el comportamiento esperado.

A continuación en la Figura 4.16 se presentan, a modo de resultados, las gráficas generadas por los experimentos simulados, medidos dentro del LBV.

4.9. Ventajas del sistema de manipulación gráfica de fenómenos físicos reales, asistidos por computadora

La comparación del *sistema de manipulación gráfica* con respecto a otros simuladores de Física, ha servido como pauta para hablar de sus ventajas no sólo como un simulador de eventos físicos que se complementa con el *LBV* y *la enciclopedia de la Física*, sino también como una herramienta que puede utilizarse de manera independiente.

Las escenas de los experimentos del sistema están construidas a partir de formas tridimensionales básicas que simplifican el fenómeno real, centrando la atención del alumno en el comportamiento esencial del fenómeno y no en lo aparatosas que pueden llegar a ser las reacciones de los objetos involucrados, al menos en los fenómenos relacionados con la mecánica clásica, este factor es irrelevante.

El sistema permite la ubicación de los objetos en tres dimensiones, que es el modo en que éstos se observan en la realidad y por lo tanto el alumno será capaz de relacionarlos con mayor facilidad con alguna experiencia previa. La observación de un movimiento desde diferentes ejes mediante la manipulación de cámaras, complementa la información que se representa en los modelos matemáticos.

Al manipular los parámetros del experimento, el alumno podrá analizar

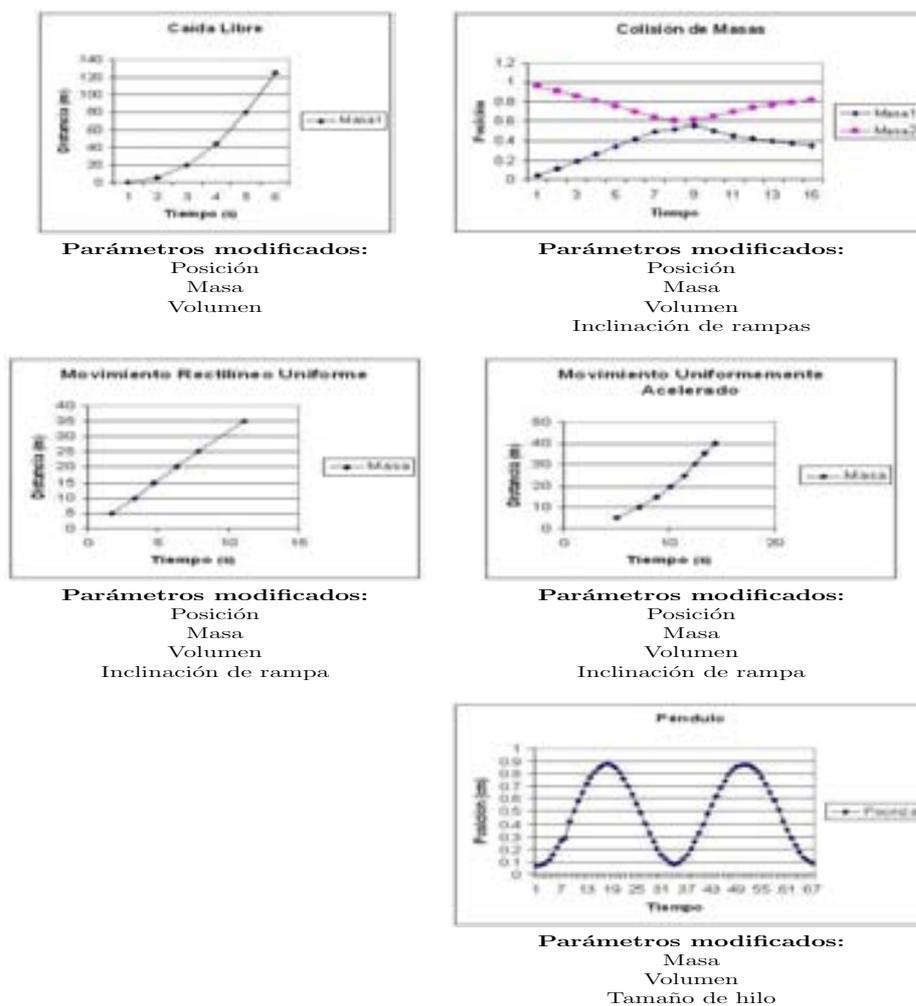


Figura 4.16: Gráficas de comportamiento de las simulaciones

el mismo fenómeno bajo diferentes circunstancias, logrando entender el comportamiento general del fenómeno independientemente de las variaciones de parámetros.

El tipo de simulación que se hace con Havok, permite tomar en cuenta variaciones significativas en las fuerzas físicas que afectan el entorno virtual a lo largo del tiempo, consiguiendo simulaciones más acertadas y apegadas a

la realidad.

Al marcar el recorrido de los objetos dinámicos, es posible notar, al menos a simple vista, las variaciones de velocidad como consecuencia de las colisiones, de los factores físicos, etcétera.

Finalmente, gracias a que el sistema permite generar videos de los experimentos, el alumno podrá conservarlos para medirlos dentro del LBV y compararlo con otros del mismo fenómeno, ya sean reales o virtuales.

4.10. Comentario final

La creciente necesidad de lograr que los estudiantes relacionen los aspectos teóricos con los prácticos de un curso de Física ha impulsado el uso de simuladores de fenómenos físicos para apoyar el proceso de aprendizaje alcanzando resultados satisfactorios pero, en el caso de México, la mayoría de las herramientas disponibles no cubren las necesidades de los alumnos, ya sea por estar en idiomas distintos, por estar fuera de su alcance, o por no permitirles controlar la simulación. Sin embargo, a lo largo de éste capítulo se ha resaltado la posibilidad de desarrollar herramientas que satisfagan dicha necesidad aprovechando la tecnología disponible para tal fin, propiciando que el alumno no se dedique únicamente a tomar notas sino que sea capaz de decidir las circunstancias bajo las cuales se desarrollarán las simulaciones.

El *sistema de manipulación gráfica de fenómenos físicos reales, asistidos por computadora* es una herramienta que facilita el estudio de fenómenos físicos de una manera autónoma, sencilla y práctica, ajustándose al ritmo de aprendizaje de cada alumno. A raíz de la proliferación de herramientas eficaces de este tipo, los estudiantes están más acostumbrados a incluirlas como una forma de estudio lúdica adicional al apoyo proporcionado en clase, por lo tanto se espera que el sistema de manipulación gráfica reciba una aceptación entusiasta.

Cada uno de los experimentos presentados fue previsto para que los alumnos pudieran realizar simulaciones de manera sencilla y en pocos pasos, asegurando en lo posible su atención en los fenómenos estudiados y sin problemas en el manejo del sistema; por otro lado, para asegurar su buen funcionamiento

to, es necesario hacer un seguimiento de las dificultades que puedan encontrar los alumnos en la práctica con los experimentos para hacer los ajustes pertinentes, entre ellos, lograr una forma más eficiente para disminuir el número de tareas que deben realizar para la elaboración de un video.

Conclusiones

El auge que la enseñanza asistida por computadora ha ido ganando a lo largo de los años, se debe en gran medida a que es posible acceder a diversos recursos tecnológicos que facilitan el desarrollo de materiales didácticos. Sin embargo, favorecer el aprendizaje de los alumnos sin que el manejo de las herramientas didácticas represente un problema para ellos, depende de la explotación adecuada de dichas tecnologías, de todo lo referente a la presentación de sus contenidos y del estudio detallado tanto de las características como de las necesidades del estudiante con la finalidad de cumplir los objetivos para los que estos materiales fueron creados.

La relevancia de desarrollar materiales educativos asistidos por computadora no se limita únicamente a cumplir con el objetivo de que el alumno logre entender el fenómeno que observa, refuerce sus conocimientos teóricos y los vincule con aspectos prácticos a través de la experimentación; la idea es que los alumnos puedan utilizarlos en casa sin la necesidad de trabajar en un laboratorio especializado o de la estricta supervisión de un experto. De éste modo el software educativo se adapta al ritmo de aprendizaje de cada uno de ellos y además propicia el trabajo autónomo.

A lo largo de éste trabajo, se ha hecho evidente que el desarrollo de software educativo requiere trabajar con distintas herramientas que colaboren en la construcción de componentes específicos del sistema, por ejemplo, las simulaciones, las interfaces, los videos, etcétera. En este sentido, las tecnologías multimedia han demostrado ser de gran utilidad porque son aquellas que permiten, entre otras cosas, integrar en una sola herramienta cada uno de los componentes del sistema, los dispositivos necesarios para su visualización y su control.

En el caso particular de la enseñanza de la Física, el acercamiento a la experimentación a través de simulaciones interactivas representadas en ambientes tridimensionales, permite reproducir de una manera segura y más económica, eventos físicos reales que pueden ser controlados y sometidos a diversas condiciones. Una característica importante de los simuladores es que permiten que el alumno tome un papel activo que lo motiva a indagar para descubrir tanto las causas como los efectos de los fenómenos con los que experimenta, aprendiendo de sus propios errores no a través de mensajes de error o reprimenda sino a partir del análisis de las consecuencias que sus acciones generaron en el sistema.

Particularmente, en el desarrollo del *sistema de manipulación gráfica de fenómenos físicos reales, asistidos por computadora*, el principal logro alcanzado fue la implementación de la herramienta a partir de la explotación e integración de las tecnologías disponibles más adecuadas para generar ambientes que pudieran ser controlados con dispositivos básicos que son bien conocidos por los alumnos, como el teclado y el ratón (*mouse*), logrando que el sistema cumpliera satisfactoriamente con las características requeridas:

1. Mostrar fenómenos físicos relacionados con la Mecánica Clásica.
2. Visualizar los experimentos sobre un ambiente 3D.
3. Permitir el control de parámetros.
4. Recibir una respuesta gráfica en tiempo real de la modificación de parámetros (en el caso de la posición y volumen de los objetos).
5. Generar videos de los experimentos realizados.

En el aspecto gráfico o de diseño se logró modelar los objetos tridimensionales de la simulación utilizando las primitivas predefinidas dentro de la misma herramienta de desarrollo del sistema: *Director*, con este avance se logró evitar el uso de otra herramienta especializada en modelado tridimensional gracias a que los modelos empleados fueron los más sencillos para simplificar el sistema real, además, los objetos están mapeados con una textura cuadriculada para facilitar su medición dentro del Laboratorio Basado en Video.

En el aspecto de programación, se logró construir el sistema aprovechando las ventajas que presentan Director y Havok al ofrecer piezas de código reutilizables y adaptables a las necesidades del desarrollador, entre sus aspectos más importantes, se encuentra facilitar al programador la implementación de sus propias piezas de código y transformarlas en bibliotecas independientemente de las que se ofrecen dentro del sistema, ampliando las posibilidades de construir aplicaciones más complejas a partir de elementos simples.

Por otra parte, la explicación de fenómenos físicos relacionados con la Mecánica Clásica a partir de objetos que no pueden deformarse es la más adecuada para los experimentos de caída libre, movimiento rectilíneo, colisiones y péndulo, pero también presenta algunas limitaciones de orden gráfico. Tal es el caso del experimento de *resorte-masa*, que consiste en un resorte al cual está unido una peonza. Sin embargo, únicamente aparece la peonza porque no fue posible representar gráficamente el objeto resorte, ya que era necesario mostrar en tiempo real la deformación de este objeto, de acuerdo con los cálculos realizados por el propio sistema, resultando ser un factor costoso computacionalmente hablando, dada la cantidad de operaciones que era necesario resolver para obtener visualmente una simulación apegada a la realidad del comportamiento únicamente del resorte, por lo tanto, en éste experimento se deja expresado el comportamiento del sistema bajo la influencia de un resorte, pudiendo visualizar la masa. A pesar de estas limitaciones gráficas, los resultados obtenidos en este experimento son muy similares a los reales.

Si bien han quedado cubiertos los temas relacionados con la Mecánica Clásica, aún es necesario realizar algunos ajustes en el modo de interacción con los usuarios para que sea más sencillo; para ello será necesario realizar una serie de pruebas donde los alumnos darán su opinión respecto al sistema y a las dificultades para manejarlo y entenderlo. También será necesario hacer pruebas referentes al manejo de la ayuda dentro del sistema ya que se espera que éste sea utilizado por los alumnos en su propia casa. Otro factor que se modificará es la manera en que el estudiante creará el video de la simulación para que no sea necesario abrir el constructor de películas sino que se pueda acceder a éste mediante alguna liga o botón dentro de la herramienta y su funcionamiento sea automático. Por otro lado, se analizará la posibilidad de poder integrar la simulación de un resorte donde éste pueda visualizarse gráficamente para ofrecer a los alumnos una idea más clara del

comportamiento de dicho experimento. Todos los ajustes antes mencionados deberán hacerse con la finalidad de obtener mejores resultados que se reflejen en el aprendizaje de los alumnos.

Bibliografía

- [1] J. L. Alonso, D. Gutiérrez, V. López and J. Torrecilla, Internet y Educación, Universidad de Castilla La Mancha, 1998, [http://www.uclm.es/profesorado/ricardo/WEBNNTT/...](http://www.uclm.es/profesorado/ricardo/WEBNNTT/...Bloque%202/Internet.htm#8) ...Bloque%202/Internet.htm#8
- [2] J. Arruda, Atuña, J., Un Sistema Didáctico para la Enseñanza-Aprendizaje de la Física, Revista Brasileira de Ensino de Física, 23 (2001) http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0102-47442001000300011&lng=es&nrm=iso&tlng=es
- [3] J. Banks, Carson, J., Nelson, B., Discrete - Event System Simulation, Prentice Hall, 2001.
- [4] C. Belloch, Aplicaciones multimedia interactivas: clasificación, Unidad de Topología Educativa. Universidad de Valencia., España, 2002.
- [5] J. M. Cabanellas Becerra, Introducción a los Simuladores, Escuela Técnica Superior de Ingenieros Industriales, Universidad Politécnica de Madrid, 2006, [http://gigserver.gig.etsii.upm.es/...](http://gigserver.gig.etsii.upm.es/...jmcabanellas/Doctorado_0405/Doctor_Tecno_Simul_0405_guia_4.pdf) ...jmcabanellas/Doctorado_0405/Doctor_Tecno_Simul_0405_guia_4.pdf
- [6] CERI, Information Technologies in Education: The quest for quality software, OECD - Organization for Economic Co-operation and Development, and CERI - Centre for Educational Research and Innovation, París, 1989.
- [7] C. Clips, Crocodile Physics, Crocodile Clips, 2007, [http://www.crocodile-clips.com/...](http://www.crocodile-clips.com/...Our_Products/Physics/Crocodile_Physics/) ...Our_Products/Physics/Crocodile_Physics/

- [8] T. de Jong, van Joolingen Wouter, *Discovery Learning with Computer Simulations of Conceptual Domains*, University of Twente, Holanda, 1996,
- [9] director-online, Havok Developers Site Archive, Havok, 2004, <http://director-online.com/havok/index.html>
- [10] E. England and A. Finney, *Managing Multimedia. Project Management for Interactive Media*, Addison Wesley, 1999.
- [11] FileXtra4, FileXtra4, 2003, <http://homepage.mac.com/klkersten/xtras/FileXtra4/index.html>
- [12] A. Galvis, *Ingeniería de software educativo*, Ediciones Uniandes, Bogotá, 1992.
- [13] F. Gamboa, Pérez Silva J. L., Lara Rosano F., F. Cabiedes Contreras and A. Miranda Vitela, A student centered methodology for the development of a physics video based laboratory, *Elsevier / Interacting with Computers*, 13 (2001), pp, 527-548
- [14] F. Gamboa, Pérez, J. L., Caviedes, F., Lara Rosano, F., Kemper Valverde, Miranda Vitela, A., Juárez Garduño, R., *Specification and Development of a Physics Video Based Laboratory for High School level*, *Advances in Multimedia and Distance Education*, 1999.
- [15] F. Gamboa, Pérez, J. L., Caviedes, F., Viniegra, A., *¿Qué debemos entender por Software Educativo Centrado en el Alumno? Encuentro de Multimedios Universitarios: hacia la asociación*, 2001.
- [16] M. Gándara, *¿Qué son los programas multimedios de aplicación educativa y cómo se usan? Una introducción al modelo "NOM"*, (1999)
- [17] I. García, *Simulación Híbrida como Núcleo de Simulación de Aplicaciones Gráficas en Tiempo Real*, Departamento de Informática, Universitat de València, España, 2004, http://www.tdx.cesca.es/TESIS_UV/AVAILABLE/TDX-0613105-172930/garcia.pdf
- [18] A. Guasch, et al., *Modelado y simulacion: aplicacion a procesos logísticos de fabricacion y servicios*, Universitat Politecnica de Catalunya, 2002.

- [19] W. D. Haddad, Educación para todos en la era de la globalización. el papel de la tecnología de la informática, in C. d. Moura, ed., La educación en la era de la informática, Banco Interamericano de Desarrollo, 1998.
- [20] C. Harrell, Ghosh, B., Bouden Jr., Royce, Simulation using ProModel, Mc Graw Hill, 2004.
- [21] Havok, HAVOK Physics Primer Version 1.0, Havok.com Inc, 2001, <http://director-online.com/havok/index.html>
- [22] InteractivePhysics, Interactive Physics, MSC Software Corp, 2004, <http://www.addlink.es/productos.asp?pid=79>
- [23] R. Kalawsky, Human Factors Aspects of Virtual Design Environments in Education: Project Report, Advanced VR Research Centre, Loughborough University of Technology, Manchester, 2000, pp. 145, www.jisc.ac.uk/uploaded-documents/jtap-048.pdf
- [24] H. A. Kofman, Modelos y simulaciones computacionales en la enseñanza de la física, Facultad de Ingeniería Química de la Universidad Nacional del Litoral, Argentina, 2000, pp. 10, <http://ar.geocities.com/hugoakofman/download/documentos/...modelos.pdf>
- [25] F. Lara Rosano, F. Gamboa Rodríguez, J. L. Pérez Silva, N. Kemper Viverde, J. Barojas Weber, G. Sierra, A. Miranda Vitela, R. Pérez y Pérez and F. Caviedes Contreras, Sistema inteligente computarizado para el aprendizaje conceptual e interactivo de la física a nivel bachillerato, Universidad Nacional Autónoma de México, 2000.
- [26] P. Marquès G., Multimedia educativo: clasificación, funciones, ventajas e inconvenientes, Facultad de Educación, Universidad de Barcelona, Departamento de Pedagogía Aplicada, 2004, <http://dewey.uab.es/pmarques/funcion.htm>
- [27] P. Marquès, El Software Educativo, Universidad Autónoma de Barcelona, 2006.
- [28] C. Morales, Modelo de evaluación de software educativo, Instituto Latinoamericano de Comunicación Educativa, 2000,

- [29] T. Naylor, J. Balintfy, D. Burdick and K. Chu, Técnicas de simulación en computadoras, Limusa, 1982.
- [30] Newton, Newton Explore Physics, DesignSoft, Inc., 2006, <http://www.dsmm.net/English/newton/index.php>
- [31] NTNU, Laboratorio Virtual de Física, National Taiwán Normal University para la Universidad Federal de Santa Catarina, 2005. <http://www.fsc.ufsc.br/ccf/parcerias/ntnujava/index-port.html>
- [32] J. Parra Márquez, R. García Alvarado and I. Santelices Malfanti, Introducción Práctica a la Realidad Virtual, Universidad del Bío-Bío, Chile, 2001.
- [33] C. D. Pedgen, Shannon, E. R., Sadowski, R. P., Introduction to simulation using SIMAN, Mc Graw Hill, 1995.
- [34] Pérez Silva J. L., Miranda A., Garcés A. and L. N., Hipertexto de Física, Centro de Instrumentos, UNAM, México, 1998,
- [35] J. L. Pérez Silva, Gamboa Rodríguez F., Caviedes Contreras F., Sistema de manipulación gráfica de fenómenos físicos reales, asistidos por computadora, Congreso Latinoamericano de Multimedios Universitarios, 2002,
- [36] Phun, Phun VRlab, Suecia, 2008, <http://www.vrlab.umu.se/research/phun/>
- [37] M. Presnell, Simulación por computadora en Manual del Ingeniero Industrial, McGraw-Hill, México, 1996.
- [38] D. Ríos, Ríos Sixto, Marín Jacinto, SIMULACIÓN Métodos y Aplicaciones, Ra - Ma, 2000.
- [39] F. Rodríguez, Técnicas y modelos de simulación de sistemas, Instituto Politécnico Nacional, 1991.
- [40] SEP, Programa Enciclomedia, 2006, www.sep.gob.mx
- [41] SHARP-Software, Sharp Export Xtra, SHARP Software, 2003, <http://www.sharp-software.com/products/#sharpexport>

- [42] Shockwave, Macromedia Shockwave Player support FAQ, Adobe, 2005.
http://kb.adobe.com/selfservice/viewContent.do?externalId=tn_15508&sliceId=2, <http://www.adobe.com/products/shockwaveplayer/>
- [43] F. J. Sierra, La tecnología informática y la escuela, Ciberhabitat, 2001,
www.ciberhabitat.gob.mx
- [44] UVA and U. d. Valladolid, Laboratorio Virtual, Universidad de Valladolid, 2007, <http://ps1.eis.uva.es/Simulacion.asp>
- [45] J. M. Zamarro, Núñez, María J., Molina, Gregorio, Aprender construyendo simulaciones, in TICEC'05, ed., Departamento de Física, Facultad de Química, Universidad de Murcia, 2005.
http://colos.fcu.um.es/TICEC05/TICEC05/09_556.pdf
- [46] F. Zambrano, Multimedia, 1999.
- [47] J. Zambrano and E. García, Multimedia: Traductor de funcionalidades pedagógicas, IV Congreso RIBIE, Brasilia, 1998.2007
- [48] B. P. Zeigler, Herbert Praehofer, Tag Gon Kim, Theory of modeling and simulation: integrating discrete event and continuous complex dynamic systems, San Diego: Academic, 2000.
- [49] P. Zuluaga, Multimedia, Wikipedia, 2007,
<http://es.wikipedia.org/wiki/Multimedia>