

Micromundos en la escuela y simulaciones en la universidad

Eduardo de Antueno

Universidad Tecnológica Nacional
Facultad Regional Buenos Aires- UDB-Física-Campus
Laboratorio de Multimedia y Simulaciones de Física
Buenos Aires – Argentina
Tel. (54-11)4581-2768 (54-11)15-5001-2768
eantueno@yahoo.com
info@fundaustral.com.ar
www.fundaustral.com.ar

Resumen - Abstract

El autor ha trabajado con la herramienta informática con modalidad Logo con alumnos de 9 a 14 años en distintas instituciones educativas desde 1983 hasta 1998. Actualmente desarrolla simulaciones de Física para estudiantes del nivel universitario.

Este trabajo expone, en la primera parte, las razones de utilizar esta metodología en nivel primario o EGB.

Luego, presenta simulaciones de Física para nivel terciario y universitario que ha producido en los últimos años con el lenguaje Helios®, intérprete éste, que ha sido desarrollado por un grupo de I+D, del que participó durante los '90.

Palabras clave - Keywords

*Computadoras en educación básica – Utilización pedagógica de la informática – Constructivismo
Construccionismo - Simulaciones – Micromundos – Microcosmos*

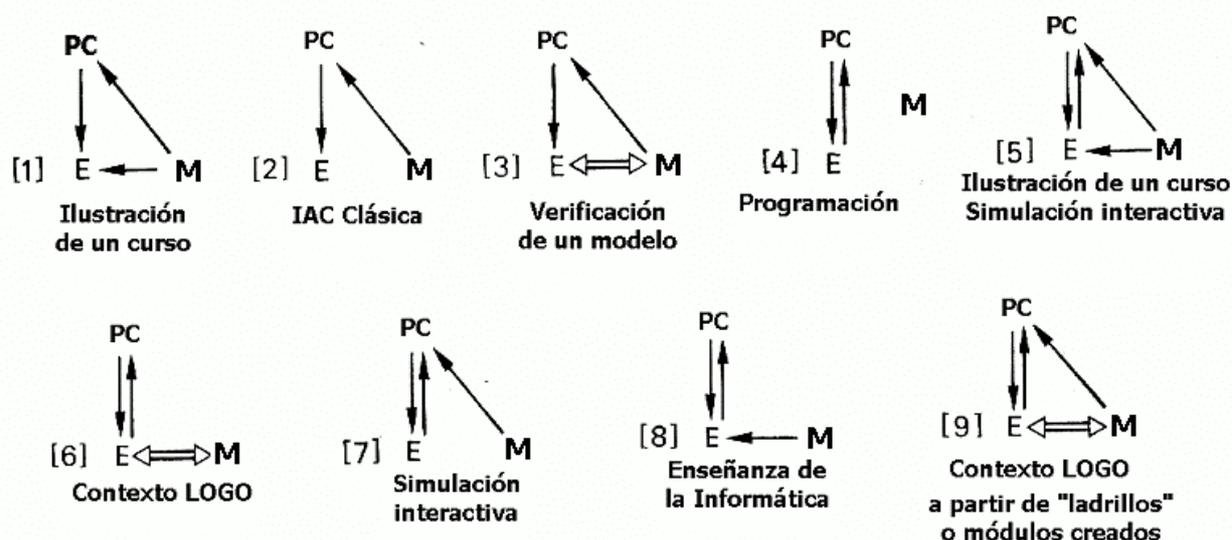
- A actúa sobre B,
- B actúa sobre A
- A y B interactúan mutuamente

A → B
 A ← B
 A ↔ B

Entonces, existen entre los 3 actores nombrados (A, M y PC) en total $4^3 = 64$ posibilidades distintas. Si eliminamos las que no tienen importancia para nuestro estudio quedan los nueve esquemas que vemos en la figura.

Modos de interactuar entre el educando E, la PC y el docente o mediador M

*Según Gérard Bossuet
 Université de Paris VI*



En el caso (2) se trata de la Instrucción Asistida por Computadora (IAC) también conocida por Enseñanza Asistida por Ordenador (EAO) aplicación “skinneriana” modernizada. Las variantes (3), (5) y (7) corresponden a simulaciones y la opción (6) es el llamado entorno de aprendizaje, micromundo o contexto LOGO, mientras que el (9) se refiere a la creación de módulos o comandos creados por el docente-facilitador para un fin determinado, generalmente usado con niños pequeños..

Desde el punto educativo, y dejando de lado el caso 8, de la Enseñanza de la Informática, pues no es éste el tema que nos interesa en este trabajo-, los casos más ricos – según mi experiencia - son las **SIMULACIONES** y el **CONTEXTO** o **AMBIENTE LOGO** (Antuena et al. -1984- págs. 262 y sgts.). Veamos brevemente qué características tiene el Lenguaje LOGO.

¿Cómo es el lenguaje Logo?

El Lenguaje LOGO fue desarrollado en el Laboratorio de Inteligencia Artificial del Instituto Tecnológico de Massachusetts (MIT) en EEUU por un equipo conducido por Seymour Papert,

colaborador de Jean Piaget en Ginebra (Suiza), y fue difundido inicialmente en EEUU, Inglaterra y Francia. Aunque LOGO fue pensado para usarse en educación, al ser un lenguaje derivado del LISP, también puede ser usado para inteligencia artificial.

Logo no es solamente un lenguaje de computación o un software muy difundido en educación, sino que es también un conjunto de materiales disponibles para someter a prueba al usuario, ya sea niño, adolescente o adulto, sobre la coherencia y consistencia de sus ideas, al tratar de ordenar algo a la computadora.

Logo representa **una modalidad en el uso de las computadoras** y un instrumento para la exploración de ideas. Se apoya en una teoría de conocimiento basada en los estudios de Piaget y explicitada por Papert (Mindstorms, 1980).

Por otra parte, su característica modular y desde el punto de vista informático, bien estructurado, lo convierte en una herramienta ideal para resolver problemas mediante técnicas de programación estructurada (Rodríguez Roselló-1984).

¿Por qué programar?

Actualmente hay una tendencia en los medios y funcionarios de educación a decir que debe buscarse que los alumnos sean usuarios inteligentes, o sea, que la computadora es una herramienta programada por otros y nosotros sólo podemos utilizar (consumir) los programas utilitarios ya desarrollados. Es bien sabido que, para manejar una computadora, no es necesario saber programar, entonces, parece que de esta afirmación - valedera por cierto - se concluye en otra afirmación: "que en la escuela está de más la enseñanza de la programación". Esta conclusión es muy discutible.

En el trabajo "¿Está obsoleta la programación?" (Clements y Meredith, 1993) se resume una reunión de la Asociación Estadounidense para la Investigación Educativa en la cual algunos importantes investigadores opinan sobre el tema. La programación es "el mejor apoyo con que contamos para representar las actividades cognitivas", según Andrea diSessa (MIT).

El investigador de la Universidad de Campinas (Brasil), J. Valente (1994) afirma: cualquier actividad intelectual puede verse como una "ventana hacia la mente". El proceso de programación como un ciclo que consiste en:

“describir → ejecutar → reflexionar → corregir → describir”.

Cuando un estudiante usa Logo sus ideas iniciales sobre cómo resolver el problema las transmite a la computadora. Así, el estudiante actúa sobre el objeto-computador.

1. Sin embargo, esta acción es también una descripción de la solución al problema mediante los procedimientos de Logo.
2. Luego la computadora ejecuta estos procedimientos.
3. El estudiante observa este proceso y el producto final y puede reflexionar sobre ellos.
4. Esta actividad reflexiva conduce a una corrección cuando el resultado es diferente a lo que el estudiante quería o si el resultado no es aceptable.

El ciclo "descripción-ejecución-reflexión-corrección-descripción" (DERCoDe) nos permite comprender el proceso de aprendizaje un poco mejor, así como por qué la programación puede ser una actividad eficaz para aprender. En este ciclo, la corrección constituye una oportunidad única para que el estudiante construya su conocimiento; para aprender sobre conceptos particulares incluidos en la solución de problemas o sobre las estrategias para la solución de problemas.

De esta manera la computadora, en un contexto apropiado, se convierte en un "espejo de la mente"(Antuena-Naveira et al.-1983) y el alumno, al pensar sobre su propio pensamiento se convierte en epistemólogo.

La modalidad LOGO no se usa para enseñar un tema determinado: es una herramienta para investigar...

El valor de la programación es independiente del contenido que se programa, da lo mismo que éste sea curricular como no curricular, académico o un juego de azar. Esta característica suele no ser comprendida en el ámbito escolar (Cavallo, 1993)

El hecho de aprender a programar no es el objetivo final sino un medio para realizar actividades de descripción de ideas y su contrastación. Se puede decir que es un microcosmos obediente que facilita el proceso de descripción de ideas.

Valor pedagógico de LOGO

El LOGO es un material formativo, como lo son los juegos de ingenio o los rompecabezas. Visto desde este punto de vista, no puede decirse de Logo que sea anacrónico ni pasado de moda, de la misma manera, que a nadie se le ocurriría decir que los rompecabezas o los juegos para armar lo son. Al igual que en un mecano, lo importante es el proceso de armado y no el producto final logrado, el camino y no la meta. Los conceptos pedagógicos incluidos en Logo permiten:

- Plantearse y resolver problemas de su propio interés...
 - a. Proponiendo soluciones.
 - b. Descomponiéndolos en partes simples (análisis).
 - c. Construir la solución total a partir de pequeñas soluciones (síntesis).
- Desarrollar la capacidad de describir.
- Explorar y descubrir.
- Valorar las componentes del lenguaje.
- Desarrollar la capacidad de abstraer.

Un concepto básico a conocer como docentes al trabajar con Logo es el de "micromundo". Un micromundo es una porción de la realidad o entorno intelectual que permite aprender interaccionando con él. Según Luis Rodríguez Roselló (1984), "un micromundo es un subconjunto de la realidad, o de una realidad construida, cuya estructura es acorde con un mecanismo cognitivo determinado, y puede suministrar un entorno donde éste último puede operar de forma efectiva.

Un procedimiento en Logo (subrutina), consiste en "enseñarle" a la máquina cómo hacer algo, y es, desde el punto de vista constructivista, la formalización de un conocimiento y su solución. Esta formalización es operacional, es decir, se puede probar, ejecutar y verificar. Desde el punto de

vista de la computadora, un procedimiento es un conjunto de frases inteligibles y ejecutables. Construir un procedimiento, para el educando, no es sólo dar un nombre a una serie de instrucciones: es un pasaje inicial a la abstracción y es también la manipulación concreta posterior a una idea (Reggini, H. C., 1983).

Cuando programa, el niño se transforma en el maestro de la máquina y no en su alumno. Convertir al alumno en maestro es un hermoso ideal pero además en una eficaz manera de que éste aprenda.

La programación como la enseñanza es una descentración, es decir que el sujeto que enseña además del conocimiento del objeto debe conocer el "lenguaje" del que aprende y debe expresarse en ese lenguaje. Conocer el lenguaje del "otro" es esencial para poder comunicarse. Este juego de "centraciones" y descentraciones es uno de los más interesantes y ricos aspectos de la programación.

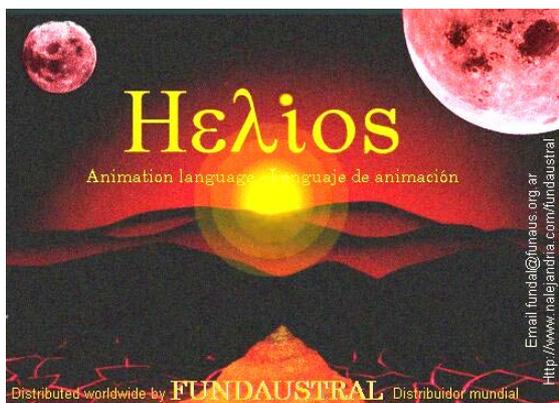
Seymour Papert crea el término "construccionismo" definido por él mismo como un movimiento piagetiano donde se agrega la componente afectiva. En este contexto ésta está representada por la "voluntad de aprender" que puede ser reforzada por la valoración que le da el individuo al objeto a conocer. Esta valoración está determinada por el "poder" que da el objeto y la "resonancia cultural" del mismo.

El autor de este trabajo ha trabajado coordinando equipos de docentes en distintos establecimientos educativos de nivel primario y secundario (edades entre 9 a 14 años), desde 1983 hasta 1998 en calidad de profesor o coordinador de informática con esta modalidad de trabajo. Se inició en el Colegio Nacional de Buenos Aires, dependiente de la Universidad de Buenos Aires, con una de las primeras experiencias en nivel medio, única por su masividad (800 alumnos) formando parte de un equipo interdisciplinario con otro ingeniero y una psicóloga con dedicación exclusiva. Esa experiencia fue descrita en el libro Computación en el Colegio (1984). Luego se continuó con experiencias en 5 colegios más de la ciudad de Buenos Aires y sus alrededores, en los cuales se trabajó con esta modalidad y lenguaje.

El nuevo desarrollo: HELIOS

Antes de aparecer las computadoras personales las primeras versiones de LOGO tenían alguna capacidades (de sonido y animación) que, con el advenimiento de la PC, se perdieron.

Es así que, con la voluntad de desarrollar una versión mejor que las existentes y que brinde



mayores posibilidades de programar animaciones, se integró en Fundaustral (ONG dedicada a difundir la UPI) un grupo de I+D formado por el Ing. Carlos Grant, Eduardo de Antueno e Ignacio Luppi que a principios de los '90 concretó la versión de Logo Gráfico, primero para DOS y luego para entorno Windows®, traducida luego al inglés (Graphic Logo) y portugués, esta última ampliamente difundida en Brasil bajo el nombre de Trend-Logo.

La misma fue presentada en eventos

internacionales en Cuba, EEUU, México, Hungría, Brasil, Chile, Uruguay y Argentina y utilizada por más de 200.000 alumnos en Brasil y Argentina (ver www.fundaustral.com.ar y www.nalejandria.com/fundaustral)

Luego, en el período 2000-2005 se desarrolla la primera versión de HELIOS, mucho más potente y con mayores recursos de multimedia que las anteriores para Windows 9x/XP/Me. Los materiales de que dispone son:

- a- Un elemento graficador (con apariencia de tortuga) que construye dibujos planos y tridimensionales desde una visión intrínseca, es decir con desplazamientos y giros alrededor de 3 ejes. La tortuga, matemáticamente hablando, es un vector, pues en todo momento tiene ubicación, dirección y sentido. Al comandar la tortuga tratando de dibujar algo el operador debe descentrarse, ubicándose en el lugar de ésta, por lo que las líneas se construyen a partir de referencias concretas de su propio cuerpo. Utiliza la llamada geometría intrínseca.



- b- 120 actores o formas móviles que permiten crear animaciones para representar cualquier tipo de fenómeno, historia o recrear experiencias físicas en la pantalla.



- c- Imágenes fijas de cualquier tamaño (con extensiones BMP, PCX o JPG).
- d- Sonidos y archivos de video del tipo WAV, MIDI o AVI.
- e- Manejo de dispositivos externos conectados a la computadora como interruptores, motores, sensores físicos, ópticos o térmicos, etc.
- g- Lugar donde almacenar la información a procesar. La forma en que LOGO guarda la información - y luego dispone de ella - es a través de objetos llamados "palabras" y conjuntos ordenados llamados "listas". El manejo de este tema es el que posibilita el real dominio del lenguaje.
- h- Botones de distintos tipos que permiten activar procedimientos, y controles (sliders) que pueden manejar datos numéricos.

En este nuevo desarrollo se han incluido comandos para simular entornos físicos que pueden utilizarse en temas de dinámica y cinemática con velocidad, aceleración, masa, fricción, aceleración, gravitación, movimientos oscilatorios armónicos con o sin amortiguamiento. Además se diseñó un editor de imágenes propio que facilita la creación de efectos visuales como la rotación y alejamiento de objetos (ver figura).

También detecta colisiones y otros conceptos que facilitan introducirse en el mundo de las simulaciones a nivel terciario, como describo en la segunda parte.

▪ **PARTE 2: Su uso actual en la universidad**

La inclusión de un microcosmos de Física es una característica única en el mundo respecto de otras versiones de intérpretes de Logo permite desarrollar simulaciones para su uso educativo a nivel universitario.

En los países de Europa del Este ha habido en los '90 un interés creciente en el uso de Logo no solamente a nivel primario sino a nivel universitario para la exploración de las matemáticas y la física. Grupos de investigación se ocupan del tema en distintos países, como en EEUU: Boston University, MIT, Israel: Bar Ilan University Department of Math and Computer Science, Rusia: Institute of New Technologies of Education, Eslovaquia: Comenius University Department of Informatics Education, Bulgaria: Sofia University, Faculty of Mathematics and Informatics, Yugoslavia: University of Novi Sad, Institute of Mathematics, y en otras universidades en Inglaterra, Hungría, Eslovaquia, Grecia y Portugal.

Por estos motivos el autor inicia el desarrollo de simulaciones en los '80 con las primeras versiones de Logo como la versión de Texas Instruments TI 99/4A, que son publicadas en 1984 y una decena más, publicadas en memorias de congresos del área de IE, y a partir de 2004 en la Universidad Tecnológica Nacional en el Laboratorio de Multimedia y Simulaciones de Física (UDB Física).

Fundamentación de un proyecto

Actualmente los cronogramas de estudio de Física suelen ser muy ajustados, los docentes cuentan con poco tiempo para desarrollar cada tema, y los alumnos, en muchos casos, realizan menos experiencias de las deseadas para internalizar y madurar los conceptos nuevos.

Es deseable contar con recursos que permitan agilizar la enseñanza-aprendizaje y brindar a los estudiantes un entorno experimental accesible. Uno de estos recursos, es la modalidad de "simulación" que provee un entorno adecuado. Como ha sido expuesto en innumerables congresos, difícilmente los docentes utilicen un software que no se ajuste exactamente a sus necesidades, por lo tanto resulta pertinente encarar el desarrollo de software específico aplicable a las necesidades de las cátedras. Dicho material debe cumplir con 4 características: ser pertinente, adecuado, sencillo y en idioma castellano y debe estar disponible tanto para clases teóricas y prácticas, y a disposición de los alumnos, en horarios adecuados y en lo posible, en Internet para ser consultado desde fuera del ámbito de estudio.

Por medio de la simulación por computadora, ciertos conceptos pueden ser mejor comprendidos, ya que esta metodología brinda una perspectiva distinta y complementaria a la vía experimental tradicional. Cuando un fenómeno es muy complejo, la simulación permite eliminar algunas variables, simplificando el modelo, y facilitando su comprensión.

Por otro lado se puede trabajar en condiciones experimentales imposibles de lograr en un laboratorio, por ejemplo, variar la aceleración de la gravedad o, más aún, experimentar variando algunos modelos teóricos aceptados y evaluar sus posibles consecuencias.

Metodología de Desarrollo

El autor, docente de la Universidad Tecnológica Nacional, Facultad Regional Buenos Aires, actualmente desarrolla software educativo por medio de simulaciones que permitirán interactuar con elementos virtuales que facilitan al alumnado la apropiación de conocimientos, y por otro lado, acortan el tiempo necesario para desarrollar algunas experiencias. También servirán para contrastar las soluciones de problemas propuestos. Dicho material desarrollado se proyecta compartirlo vía internet con otras unidades académicas que lo soliciten.

Los guiones de los futuros materiales didácticos podrán desarrollarse a partir de ideas del grupo, que se encuentra en formación, o de otros docentes de la cátedra que lo soliciten. También se solicitará la participación de pedagogos y profesionales de la comunicación.

El desarrollo también tendrá en cuenta aspectos lúdicos y un enfoque pedagógico basado en el constructivismo. Se harán sesiones de "brainstorming" alternando momentos de creatividad totalmente libre, donde se aporten ideas generales sobre el programa, con otros momentos donde la actividad creadora se vaya concentrando en la concreción de las características específicas que configurarán el diseño funcional. Se pretende lograr un producto de calidad que cumpla con las características de funcionalidad, usabilidad y fiabilidad, características éstas deseables y necesarias para un material educativo interactivo (Díaz et al. 2003)

En 2005 iniciamos una etapa piloto, con 6 aplicaciones piloto en Cinemática y Óptica desarrollados por el autor de este trabajo en el Laboratorio de Simulaciones y Multimedia, que se puso a disposición de los docentes. Este año, en la UDB Física, se realizaron talleres de actualización para docentes en distintos temas, que incluyeron el uso de simulaciones.

Etapas de las actividades y acciones previstas

a) **Conceptualización y Pre-producción (Génesis y Prediseño)**

Durante esta etapa el grupo genera las ideas, teniendo en cuenta las dificultades de aprendizaje de los distintos contenidos.

Se crea la propuesta y se estudian las diferentes alternativas, en base a las posibilidades de la NTI para proponer soluciones didácticas. Duración estimada 8 meses.

b) **Producción:** En esta etapa se materializa la propuesta en elementos perceptuales, metodológicos y técnicos: Programación, elaboración del prototipo-alfa, y evaluación interna y piloto externa.

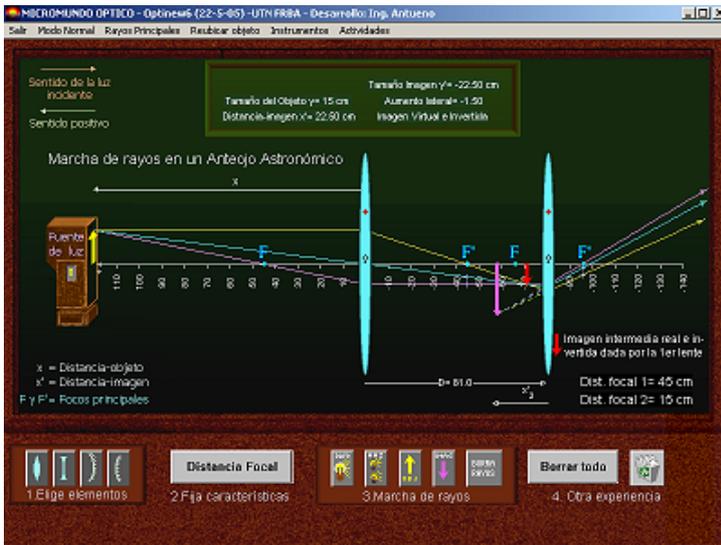
Se genera el guión en el cual se formaliza y consolida la idea a materializar. Luego se pasa a realizar dibujos y fotografías digitales o escaneados y se va desarrollando el programa.

Impacto: Durante esta etapa un prototipo será evaluado por docentes ajenos al grupo, ya que es de suma importancia su aceptación. Duración estimada 16 meses.

c) **Post-producción:** En esta etapa se realizan pruebas con los usuarios (estudiantes y docentes), con el fin de realizar los análisis correspondientes al producto vs. el público usuario, proponiendo y desarrollando los necesarios ajustes, la depuración del software y la publicación en Internet. Duración estimada 8 meses.

d) **Mantenimiento post-producción** – Continua.

En el desarrollo de software siempre se detectan mejoras posibles, por lo que en función del uso que le den los alumnos se hace necesario un mantenimiento, además de posibles cambios a introducir por la evolución del software de base.



Material desarrollado

Micomundo óptico

Consiste en la simulación de un banco óptico para desarrollar experiencias.

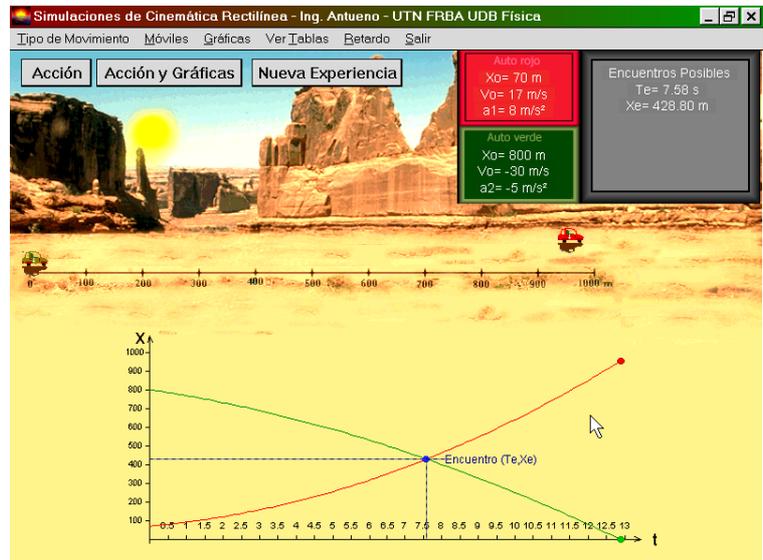
Permite colocar lentes delgadas y espejos experimentando con varios elementos simultáneamente (Sistema Óptico Centrado) pudiendo obtener todos los resultados que se precisan en la asignatura Física al nivel universitario.

Cinemática Rectilínea

Consiste en una simulación de 2 móviles con MRU o MRUV que al desplazarse permiten graficar X vs T y V vs T .

Es posible fijarle a ambos las condiciones iniciales de x , v y a , obteniendo también las tablas de sus posiciones o velocidades cada décima de segundo.

En caso de producirse 1 ó más encuentros los indica en la gráfica y calcula los valores mostrándolos en la pantalla.

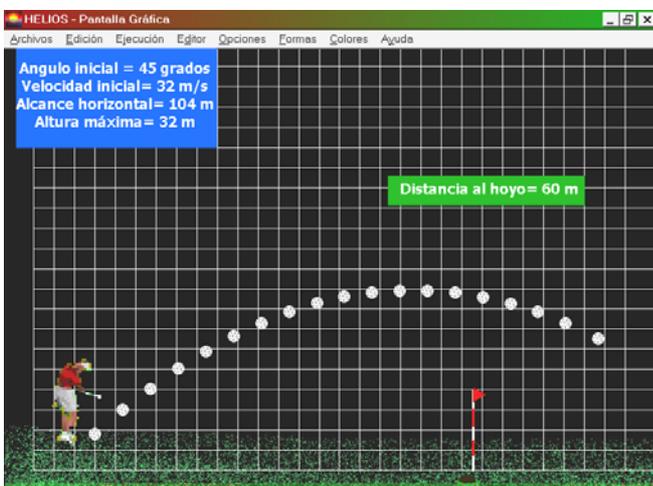


Es posible modificar las escalas de las gráficas, así como la velocidad de la simulación.

Se encuentra en fase de utilización idem anterior.

Cinemática Bidimensional I Tiro Oblicuo - Golf

Consiste en una simulación de un tiro de golf en el que se le fija el valor inicial del vector velocidad y se trata de embocar en el hoyo que se presenta a distintas distancias.

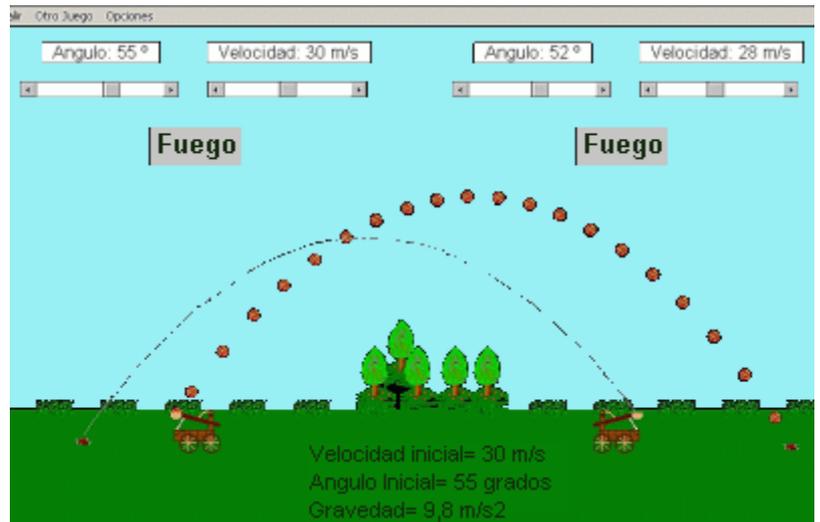


Puede verse la trayectoria de la pelota o bien optar por una secuencia de fotografías estroboscópicas, como se ve en la figura.

Es posible variar parámetros como la aceleración de la gravedad, coeficiente de restitución de la pelota y rozamiento con el aire.

Cinemática Bidimensional II Tiro Oblicuo – Catapultas

Consiste en una simulación lúdica con 2 catapultas que pueden variar módulo de velocidad y ángulo tratando de derribar a la otra. En cada juego varía la gravedad del lugar para dificultar la tarea. También hay obstáculos que a veces

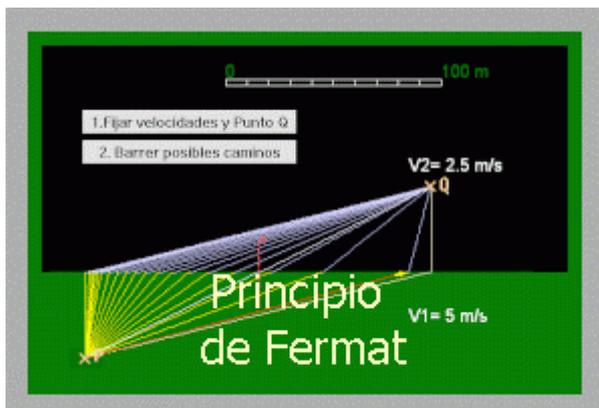


impiden disparar con determinados ángulos. Se encuentra en fase de utilización en los Trabajos Prácticos de Física 1.

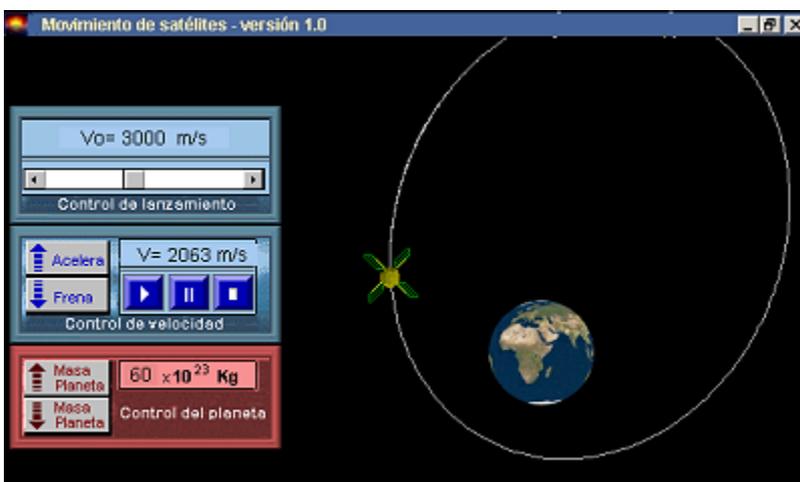
Principios de la óptica – Fermat

Se trata de una experiencia donde se hacen recorridos calculando el tiempo que tarda en pasar de un punto a otro teniendo que atravesar distintos medios. Son variables las posiciones de los puntos y las velocidades en los medios.

Se trata de verificar el Principio de tiempo mínimo de Fermat.



Otras simulaciones



Se encuentra en estado de desarrollo otras simulaciones sobre movimiento de satélites, ondas, péndulo balístico y choques.

Durante 2005 y 2006 se evaluarán los resultados obtenidos por la inclusión de estos desarrollos en las clases prácticas y en algunos casos como apoyo a clases teóricas.

Por otro lado, luego de terminar la etapa de prueba de las versiones beta, se colocarán los materiales en

la web para uso de los alumnos.

Referencias

- Antueno E. y Luppi I - *Aprendiendo con Logo Gráfico* – Ediciones Funda Austral 1997 – Buenos Aires - Argentina
- Antueno E., Naviera A. y Thompson H. -*Computación en el colegio, Logo: espejo de la mente*. Argentina, Editorial Informática Educativa(1984).
- Antueno, Eduardo – Luppi, Ignacio – Grant, Carlos - *A new Latin American Logo Versión –Learning and Exploring with Logo – Proceedings VI European Logo Conference- M. Turcsanyi Szabo Editor - 1997– pags 417 a 420.*
- Abelsson, Hal and DiSessa Andrea (1981)– *Turtle Geometry* – The MIT Press Cambridge, MA.
- Abelsson, H and Sussman, G.J. (1985) *Structure and Interpretation of Computer Programs* –
- Bossuet, Gérard – *La computadora en la escuela (L 'Ordinateur a l'école)*– PUF Francia 1982
- Clements, D H. y Meredith, J.S. ¿“Is Programming Obsolete?” *Logo Exchange*, (12) 1, Otoño 1993.
- Cavallo, Eduardo E. (3/12/93) "LA FUERZA Y LA DEBILIDAD DEL LOGO"- Conferencia de Clausura del Encuentro de Reflexión DIEZ AÑOS DESPUES - Asociación Amigos de Logo
- Díaz, M.G -Antón, M.A. Pérez, Grimán, A.- Mendoza L. (2003) "Propuesta de una metodología de desarrollo de software educativo bajo un enfoque de calidad sistémica" Presentado en el IV Congreso Multimedia Educativo, Universidad de Barcelona, España.
- Harvey, Brian - *Computing Science Logo Style* - Univ. de Berkeley – EEUU- The MIT Press Cambridge, MA.
- Harvey, B. and Wright, M. (1994) *Simply Scheme – Introducing Computer Science*, The MIT Press Cambridge, MA.
- Morfin Otero, Francisco (1993) *Los procesos educativos mediados por la computadora*, ITESO, Guadalajara, México.
- Papert, Seymour (1980) *Mindstorms: Children, Computers and Powerful Ideas*. Basic Books, New York, USA.
- Papert, Seymour “¿Is Programming A Good Activity for Children?” *Logo Update*, (2) 1, Otoño 1993 - Logo Foundation – N.Y. USA.
- Pere Marquès (1995) *Metodología para la elaboración de software educativo. Guía de uso y metodología de diseño*. Barcelona: Editorial Estel
- Piaget Jean (1967), *Logique et epistemologie scientifique* – Paris - Gallimard
- Ramírez Adriana M. Et al. - *Software Educativo- Metodología de desarrollo e incorporación en los ambientes de aprendizaje* - Universidad EAFIT y Universidad Pontificia Bolivariana - Colombia
- Reggini, Horacio C. Conferencia de apertura del 1er Congreso Internacional Logo-Computadoras y Educación, Buenos Aires, 16/9/83(ver biblioteca virtual de Funda Austral en www.fundastral.com.ar)
- Rodriguez Roselló, Luis (1984) *De la tortuga a la inteligencia artificial* - Vector Ediciones- Madrid - España
- Sendov Bojidar - Sendova Evgenia (1997) *Tuning a Logo-like environment to a knowledge domain* – Eurologo 97 Proceedins, pp.28-39.
- Solomon, C. “Introducing Logo to Children: Teaching Logo Requires an Awareness of Different Learning Styles” *Byte*(7) 8, 1992, pp. 196-208.
- Tharp, Roland G. y Gallimore, Ronald - *Rousing Minds to life - teaching, learning, and schooling in social context*, Cambridge University Press, Cambridge, 1988.
- Valente, J.A. “Computers in Education: Shifting the Pedagogical Paradigm from Instructionism to Constructionism”. *Logo Exchange*, (12) 2, 1994, pp. 39-42-Logo Foundation NY-USA.
- Watt Daniel Lynn y Watt Molly Lynn (eds.), *New paradigms in classroom research on logo learning*, ISTE, Oregon, 1993.
- Weir, S. *Cultivating Minds: A Logo Casebook*. Harper & Row, Nueva York, 1987