



ISSN 1988-6047 DEP. LEGAL: GR 2922/2007 Nº 19 – JUNIO DE 2009

“LEGO MINDSTORMS, ACERCANDO LA ROBÓTICA AL ALUMNADO”

AUTORÍA ELENA VARO MARTÍNEZ M ^a TERESA TORRES PINTOR
TEMÁTICA TIC, TECNOLOGÍA
ETAPA EDUCACIÓN SECUNDARIA

Resumen

En algunos proyectos de renovación metodológica para la enseñanza integrada de las ciencias, las matemáticas y la informática, donde la introducción de robots “didácticos” en la escuela parece que puede contribuir a realizar aprendizajes más significativos y motivadores, se plantean preguntas tales como: ¿qué interés ofrece el introducir la robótica en la educación primaria y secundaria? ¿Qué se aprende con las actividades de Robótica? ¿Cómo se diseñan experiencias de “buen aprendizaje” (constructivista) con robots?

Palabras clave

Currículum, tecnología, tic, robótica, LEGO

1. INTRODUCCIÓN

En una sociedad cada vez más desarrollada tecnológicamente, los sistemas automáticos y robóticos han salido de un ámbito puramente industrial o de investigación hacia entornos muy próximos a la sociedad en general: electrodomésticos, vehículos, sistemas de comunicaciones, etc., son sólo algunos ejemplos. Es por ello, por lo que debemos presentar estos temas al alumnado para prepararles ante un futuro, que se hace presente a pasos agigantados.

Desde el punto de vista didáctico y metodológico, trabajar con estos sistemas permite integrar de forma armónica a través de pocas actividades la mayoría de los bloques de contenidos de las materias de tecnología.

Llevar estos sistemas al aula por parte del profesorado está siempre limitado a la falta de este tipo de materiales en los centros, al excesivo coste de los materiales existentes en el circuito comercial



ISSN 1988-6047 DEP. LEGAL: GR 2922/2007 Nº 19 – JUNIO DE 2009

y a la falta de experiencia previa y formación del profesorado en lo que a los contenidos relacionados con estos temas se refiere.

2. ¿POR QUÉ UTILIZAR LA ROBÓTICA EN EDUCACIÓN?

Considerando el concepto de tecnología educativa desde una perspectiva amplia, la utilización de robots resulta un “medio” didáctico inmejorable para que el profesor implemente una enseñanza constructiva “mediante proyectos de indagación” y de resolución de problemas que se pueden trabajar en el marco curricular tanto de la educación primaria como de la secundaria. Hasta ahora la escuela, siguiendo a Piaget, ha puesto de manifiesto la importancia de la actividad manipulativa sobre los objetos concretos para construir los esquemas de las operaciones concretas. El uso de robots programables nos permite dar un salto adelante, pudiendo realizar también una actividad manipulativa sobre los objetos textuales, para construir los esquemas de las operaciones formales.

Las teorías de Piaget [Piaget 1972, 1974] y de Vygotski [Vygotski 1978] ofrecen el marco teórico adecuado para llevar a cabo estas actividades, que pueden realizarse tanto con robots virtuales, como la célebre “tortuga” de LOGO Microworlds (<http://www.microworlds.com>), como con robots reales simples, entre los que se encuentra la igualmente conocida tecnología de los robots LEGO.

Con robots virtuales, se aprende a resolver constructivamente problemas relativos a un entorno, sin interferencia de las características físicas entre la tortuga y el entorno (“ejecutor ideal”). En cambio con robots reales, se aprende adicionalmente las reglas de control del propio robot frente al entorno físico (ya no es “ejecutor ideal”).

El diseño de experiencias de “buen aprendizaje” (constructivista) con robots se debe hacer teniendo en cuenta los siguientes aspectos:

- Proponiendo a los alumnos “clases” de problemas a resolver (tareas de una misma clase)
- Cooperando, profesores y alumnos, para su resolución en la “zona de desarrollo próximo” (Vygotski)
- Integrando finalmente las clases de tareas en procedimientos técnicos y tecnológicos de carácter más general y abstracto



ISSN 1988-6047 DEP. LEGAL: GR 2922/2007 Nº 19 – JUNIO DE 2009

2. SOFTWARE DE CÓDIGO ABIERTO Y EDUCACIÓN

La definición de software de código abierto hecha por la Iniciativa de Código Abierto dice: software que debe ser distribuido bajo una licencia que garantice su derecho a leer, redistribuir y utilizar el software libremente.

La idea es que los códigos de software lleguen a conformar una red global para los desarrolladores. El software va a través de un tipo de evolución natural que resulta en un rápido desarrollo, confiabilidad incrementada y disminución de costos.

Probablemente, la pieza de software más popular que haya llegado al movimiento de software de código abierto es el sistema operativo GNU/Linux. Este sistema operativo puede encontrarse en una amplia variedad de aplicaciones, que van más allá de las estaciones de trabajo. Desde los robots humanoides a la grabadora de televisión digital TIVO®. La confiabilidad, calidad y lo asequible de su código fuente, junto con su capacidad de ajustar su complejidad de acuerdo a la aplicación, hacen de Linux un excelente candidato para la educación. Linux es único en el sentido de que es un candidato tanto para los sistemas de software robótico, como para las estaciones de trabajo de los desarrolladores. Este es, precisamente, el tipo de flexibilidad que hace de Linux y del software de código abierto en general, un excelente candidato para la robótica educacional.

El software de código abierto puede proporcionarnos una tecnología libre y de bajo costo en el aula que, de otra manera, sería inalcanzable. Además, al utilizar software de código abierto, los estudiantes pueden participar en una gran comunidad distribuida de desarrollo de software. Ellos podrán interactuar con grandes bases de códigos del “mundo real” y tomarán parte en todo proceso importante de revisión de pares.

3. SOFTWARE DE CÓDIGO ABIERTO Y ROBÓTICA EDUCACIONAL. UNA REVISIÓN

LEGO ha producido un paquete de elementos de robótica: el Sistema de Invención Robótica MindStorms, de LEGO. Este sistema consiste en una variedad de las partes LEGO tradicionales, así como sensores, motores y el “bloque” programable primeramente el RCX y posteriormente el NXT, que está diseñado para engancharse con las piezas LEGO estándares. Estos diferentes componentes se combinan para construir una plataforma robótica.

El “bloque” RCX es un microcontrolador Hitachi H8 con 16 K de ROM y 32 K de RAM. Los programas escritos en un computador, pueden traspasarse al microcontrolador, a través de un enlace infrarrojo al RCX. El programa descansa como residente en la RAM y las rutinas las corre el sistema ROM.



ISSN 1988-6047 DEP. LEGAL: GR 2922/2007 Nº 19 – JUNIO DE 2009

El bloque NXT es un microcontrolador ARM7 de 32 bit con 64k RAM cuyos programas escritos en el ordenador, pueden traspasarse al microcontrolador por bluetooth o bien por puerto USB. Tiene 4 puertos de entradas para los sensores y 3 puertos de salida para los actuadores

Mientras LEGO empaqueta su kit MindStorms con un ambiente de programación visual, muchos estudiantes pueden encontrarse utilizando otras formas de programas como C para programar su bloque RCX o NXT.

Mucho del éxito de LEGO mindstorms puede atribuirse a una comunidad que trabaja duro y que produce estas soluciones de programación alternativas, muchas de las cuales, son proyectos de código abierto.

Estas soluciones varían en su elección de la licencia de fuente abierta. Esta elección es un claro equilibrio entre el código abierto y el uso masivo del software.

El software se puede dividir en tres categorías. La primera de ellas, utiliza el software RCX o NXT estándar, licenciado. Esto permite al usuario avanzar y retroceder entre las soluciones alternativas y el ambiente de programación visual de LEGO. Los programas son compilados en el servidor al nivel de código-byte y luego transferidos al RCX o NXT para ser leídos por el intérprete código-byte.

La segunda categoría de soluciones de programación, requiere que software licenciado especial sea instalado en el brick inteligente de Lego. Esto permite más flexibilidad en la programación de dicho brick, pero impide el uso de otros tipos de programación, tal como el ambiente de programación visual de LEGO, durante el período en que esta aplicación esté instalada.

La tercera categoría de software, controla el RCX o NXT desde una estación de trabajo. El sistema MindStorms viene con un emisor/transmisor infrarrojo que se utiliza para cargar y descargar software al RCX o un puerto USB para el NXT. Sin embargo, es posible dejar el "bulto" de la computación en una estación de trabajo y transmitir comandos directamente al RCX vía el enlace infrarrojo o vía USB al NXT.

Contrario a algunas expectativas. LEGO no ha tratado de desalentar el software de terceros para sus sistemas MindStorms. Inicialmente, LEGO ignoró a la comunidad de código abierto, pero han comenzado a apoyar esta opción. De hecho, durante el desarrollo del Sistema de Invención Robótica



ISSN 1988-6047 DEP. LEGAL: GR 2922/2007 N° 19 – JUNIO DE 2009

(RIS) 2.0, lanzaron el kit de desarrollo de software al lanzamiento real del producto. Esta es una maravillosa demostración de cómo las compañías con fines de lucro realmente se benefician con el software libre. Las soluciones de software de código abierto indicadas en la figura 1, realmente incrementaron el valor del hardware RCX y contribuyeron a su éxito. Estos proyectos permiten a los estudiantes utilizar niveles de lenguaje de programación más altos en lugar de ambiente de programación visual.

4. SOBRE LA ROBÓTICA ESCOLAR

4.1. Robots

Un *robot* es un dispositivo electrónico y generalmente mecánico, que desempeña tareas... a través de un programa predefinido

A la hora de plantearnos la enseñanza y aprendizaje de la robótica escolar hay dos aspectos de los robots que son relevantes: las *acciones que realizan* y el *lenguaje* (de programación) *que las formula*.

4.2. Acciones, estados y comportamiento

Los robots realizan acciones “con finalidad”, que buscan un *objetivo* (el que ha previsto el profesor y los alumnos) y, para ello, el robot tiene un *comportamiento*, esto es, realiza una serie de acciones secuenciales dirigidas a alcanzar su objetivo.

A su vez, un robot tiene “estados” y las acciones modifican los estados de un robot, de manera que su comportamiento se puede ver igualmente como una secuencia de estados, desde un estado inicial a un estado final que reconocemos como el estado-objetivo.

Las acciones y los estados posibles de un robot están sometidas a las restricciones generales de los sistemas físicos y a sus leyes. Los robots no hacen lo que el director *quiere*, sino lo que el robot en interacción con el medio físico *puede* hacer (un robot no puede atravesar un muro que se interpone en su camino directo hacia una posición determinada...). Las acciones de los robots al igual que las acciones de los objetos físicos pueden ofrecer “*resistencia*” a los comandos de los alumnos (a sus acciones intencionales).



ISSN 1988-6047 DEP. LEGAL: GR 2922/2007 Nº 19 – JUNIO DE 2009

4.3. Formulación de las acciones

Los robots actúan autónomamente pero siguen “un programa predefinido” que formula las instrucciones que el robot interpreta para ejecutar su comportamiento. Diremos, sencillamente, que los alumnos deben previamente *formular las acciones* que ha de ejecutar el robot.

Cuando programa un robot, el alumno *actúa* igualmente, pero no en el espacio físico del robot sino en un espacio asociado, en un *espacio representacional*, donde las frases que formula son los significantes de las acciones del robot que son el referente. Debe, en consecuencia, aprender a usar un lenguaje funcional, formulando textos “bien formados” y adecuados para que el robot realice, en consecuencia “las buenas acciones” deseadas.

Al texto que ordena realizar un cierto comportamiento para alcanzar un determinado objetivo, lo denominaremos una *tarea*.

4.4. Las formulaciones condicionales

Recordemos que tanto en el lenguaje de la vida ordinaria como en el lenguaje instruccional para robots, las acciones siempre se realizan *condicionadas* a un contexto.

En contextos preestablecidos y conocidos de antemano, el programa para un robot suele ser un texto imperativo formado por una secuencia lineal de instrucciones de acción con parámetros bien definidos. Nos referiremos a él como un “*texto de comportamiento*”.

En contextos no preestablecidos, el alumno debe prever todas las posibles condiciones de contexto que puedan darse en el entorno (interactivo) del robot y vincular a cada una de ellas el comportamiento adecuado. Ello supone la producción de un pensamiento hipotético-deductivo por parte del alumno que debe reflejarse en un texto condicional que controle los diversos textos de comportamiento.

5. PONER EN PRÁCTICA UNA ENSEÑANZA DIVERTIDA.

Una forma de desarrollar estos contenidos es mediante la puesta en marcha de una liga lego. Existente ya a nivel mundial, y la cual consiste en poner en práctica todos los conocimientos adquiridos para conseguir un fin determinado como puede ser que el robot sea capaz de salir de un laberinto.



ISSN 1988-6047 DEP. LEGAL: GR 2922/2007 Nº 19 – JUNIO DE 2009

Esta estrategia también es aplicable al aula, de esta forma se puede dividir el aula en varios grupos a los que se les propondrá el mismo reto. El alumnado con recursos: internet, libros de textos, el profesor y sus propios conocimientos adquiridos deberán conseguir el fin propuesto.

Se valorará tanto el resultado obtenido como la imaginación pero también el interés que alumnos y alumnas muestren ante el reto ofrecido. No sólo se dará importancia a que se consiga a la perfección lo que se ha pedido.

Mediante el uso del Lego en el desarrollo del currículo para el área de tecnología en el bloque de contenidos de robótica conseguimos que el alumnado sienta que el trabajo y el aprendizaje es también un juego, de esta forma los alumnos y alumnas sentirán un especial interés a la hora de realizar el trabajo.

6. CONCLUSIÓN

El software de código abierto se ha probado a sí mismo que es una poderosa herramienta para soluciones en Robótica educativa. Las populares alternativas de programación para el LEGO RCX ofrecen excelentes “paquetes” para esta disciplina en muchos niveles. Se ha argumentado que sin software de código abierto, el RCX nunca habría alcanzado los niveles de uso a los que ha llegado. De forma similar, nuestros proyectos no habrían sido ejecutados de la forma en que lo fueron, sin la disponibilidad de códigos de fuente abiertos y herramientas fácilmente disponibles.

En la medida en que la Robótica Educativa se hace cada vez más popular, otras soluciones comienzan a emerger. Por ejemplo, con la emergencia de las especificaciones en tiempo real de Java, la idea de programar robots bajo este lenguaje, ya comienza a convertirse en realidad. La compañía Systronix está desarrollando actualmente la aplicación JCX, como reemplazo para el RCX [36]. Obviamente, Systronix se dedicó a observar la poderosa comunidad que se ha formado alrededor del RCX y es muy abierta en términos de su software y esquemas.

Mientras que nos concentramos en el RCX de LEGO, debemos considerar el incremento en el uso y desarrollo del software de código abierto para su uso en robótica. Por ejemplo, Linux-RT puede ser encontrado en algunos proyectos robóticos, entre los que destaca un robot humanoide que camina.

ActiveMedia, la compañía creadora de la serie Pioneer para robots móviles, ha desarrollado una biblioteca GPL (Interfaz para Aplicaciones Robóticas de ActiveMedia o ARIA), para controlar sus robots. Aria tiene una licencia doble, también bajo una licencia de propietario, permitiendo a los investigadores dedicarse a crear aplicaciones robóticas de código abierto y software de control.



ISSN 1988-6047 DEP. LEGAL: GR 2922/2007 N° 19 – JUNIO DE 2009

El simple acto de caminar hacia una sala de clases con un robot, puede cautivar lo que, de otra forma, sería una sala sin estudiantes. Realmente, el motivar a los estudiantes a programar los robots, los conecta con una serie de herramientas pedagógicas. A medida que más y más educadores reconocen el poder de la robótica educativa, deben fortalecer su opción de elegir soluciones de código abierto. Lo mejor de este tipo de software, es que es más robusto que muchas soluciones comerciales, gratuito o de costo muy bajo y estimula la investigación colaborativa y el desarrollo.

7. BIBLIOGRAFÍA

- Vygotski, L.S. (1978) *Mind in society. The development of higher psychological process*. Cambridge, Ma.: Harward University Press. Trad. Cast. de S. Furió: *El desarrollo de los procesos Psicológicos superiores*. Barcelona: Crítica, 1979.
- Piaget, J. (1972). *The Principles of Genetic Epistemology*. N. Y.: Basic Books.
- Piaget, J. (1974), *To understand is to invent*. N.Y.: Basic Books.
- CHEVALLARD, Y. (1999) *El análisis de las prácticas docentes en la teoría antropológica de lo didáctico*, Recherches en Didactique des Mathématiques, Vol 19, nº 2, pp. 221-266,
- Baum, D. (2000), *Dave Buam's Definitive Guide to LEGO® MindStorms™*. Apress,
- Baum, D., Gasperi, M., Hempel, R., and Villa, L. *Extreme MINDSTORMS™: An Advanced Guide to LEGO® MINDSTORMS™*. Apress, 2000.
- Beer, R., Chiel, H., and Drushel, R. (1999), *Using Autonomous Robotics to Teach Science and Engineering*. Communications of the ACM.
- Fagin, F. (2000), *Using Ada-Based Robotics to Teach Computer Science*. In Proceedings of ITiCSE Annual Conference

Autoría

- Nombre y Apellidos: M^a Teresa Torres Pintor y Elena Varo Martínez
- Centro, localidad, provincia: Córdoba
- E-mail: teretorres84@hotmail.com y p22vamae@gmail.com