



Programando el robot Lego Mindstorms NXT 2.0 con Java LeJOS

Lina Ernestina Arias Hernández^a, Valeria Rubí Hernández Cisneros^a y Luis Orlando Aguirre Fuentes^a

^aInstituto Tecnológico de la Laguna., elarias@hotmail.com, valeanimix@gmail.com, ludvicaguirre@gmail.com

RESUMEN

En los cursos que involucran la programación de microcontroladores y el uso de sensores, es deseable que los estudiantes prueben sus algoritmos en una plataforma que les permita visualizar y comprobar el comportamiento de los mismos. De esta manera los estudiantes pueden experimentar el comportamiento real de sus programas. En este trabajo estamos proponiendo el uso del robot Lego Mindstorms NXT 2.0 como plataforma de desarrollo, ya que cuenta con sensores y actuadores y un microcontrolador que se puede programar en lenguaje de alto nivel. El robot Lego nos provee una plataforma que puede ser programada con diversos lenguajes de programación como Java LeJOS, C#, C y ambientes de desarrollo integrados gráficos como Microsoft Robotics Studio y NXTG, por mencionar algunos. Como resultado de la evaluación de los lenguajes anteriores, se eligió al lenguaje Java LeJOS por ser un software libre y además es orientado a objetos. Se presentan la comparativa de NXTG y Java LeJOS en dos algoritmos, el primero es para que el robot recorra, de manera autónoma, una pista con obstáculos y puentes y el segundo es para que el robot resuelva un laberinto.

1. INTRODUCCIÓN

El Kit Lego Mindstorms NXT 2.0 posee elementos básicos de robótica, así como la unión de piezas y la programación de acciones de manera interactiva. Este kit se encuentra en dos versiones, la comercial y la estudiantil. La versión comercial cuenta con 619 piezas y la estudiantil con 413. En la Figura 1 se pueden apreciar algunos de los componentes del kit estudiantil.

Las partes básicas que contienen estas versiones son:

- Microcontrolador: un ARM7 de 32 bits, con 256 KB de memoria Flash y 64 KB de RAM.
- Puertos: el bloque NXT se localizan 4 puertos para los sensores y 3 para los motores.
- Comunicación: El bloque puede comunicarse a una computadora mediante una interfaz USB versión 2.0, además de poder comunicarse con diferentes dispositivos a través de una interfaz Bluetooth, ya sea comunicación con otros robots, computadora o algún dispositivo móvil de las cercanías.
- Piezas de construcción.
- Sensores: Las diferentes versiones proporcionan diferentes tipos de sensores como pueden ser de luz, de sonido, de color, de contacto y ultrasónico.

En la Figura 1 se muestra al ladrillo NXT con sensores y servomotores.

Se realizó una evaluación entre diversas plataformas de desarrollo y se seleccionaron dos: NXT-G y Java LeJOS. La primera es el entorno de desarrollo gráfico que viene adjunto y la segunda es un lenguaje de programación orientado a objetos. Ambas plataformas tienen su nivel de complejidad, porque a pesar de que el NXT-G sea un lenguaje de bloques previsto principalmente para niños desde los 8 años, muchas de las tareas o algoritmos a ejemplificar por los estudiantes llevan un



nivel de lógica algo compleja para la simplicidad del mismo, mientras que el Java LeJOS al ser un lenguaje orientado a objetos les permite a los estudiantes de ingeniería el desarrollo de cualquier tipo de algoritmo factible de probar en el robot.



Figura 1. Componentes del Lego Mindstorms

2. HERRAMIENTAS DE DESARROLLO

Actualmente existe una gran cantidad de lenguajes (tanto libres como propietarios) para programar el LEGO Mindstorms NXT. Inclusive han surgido ambientes de programación alternativos al LEGO NXT-G que permiten una programación más avanzada: como NXC (Not eXactly C) y el RobotC que son herramientas de programación estructurada, emplean un lenguaje similar al C; y por otro lado está el Microsoft Robotics Studio, entorno de desarrollo en bloques (similar al NXT-G), y además, herramientas de desarrollo orientadas a objetos como el Java LeJOS NXJ (Lego Operating System) o C#.

2.1 Entorno gráfico NXT-G

El NXT-G es un ambiente gráfico de desarrollo que emplea bloques para el control de los robots y la captura de información desde los sensores. En la Figura 2 se muestran las partes que conforman dicho ambiente: tiene las paletas para bloques, un área de programación y ejecución, un área para establecer las propiedades de cada bloque y una guía interactiva para la construcción y programación de varios robots.

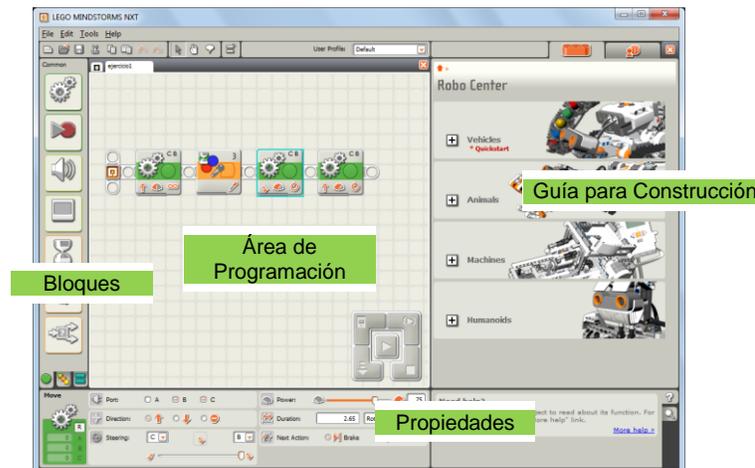


Figura 2. Entorno de desarrollo NXT-G.



El NXT- G es el software principal con el cual viene el robot Lego Mindstorms, este es un ambiente gráfico, con el cual se pueden realizar procesos paralelos o hilos, dando así diferentes respuestas con los motores o sensores, además de permitir el control con Bluetooth. El NXT-G permite programar, compilar y descargar un programa al “ladrillo” NXT para posteriormente ser ejecutado. Incorpora tutoriales y herramientas de actualización de firmware.

2.2 Lenguaje de Programación Java LeJOS NXJ

Una de las ventajas de Java LeJOS NXJ con respecto a alguno de los lenguajes anteriormente mencionados es que es una plataforma completamente gratuita (Freeware) y se puede desarrollar para iOS X, LINUX y Windows.

También cabe resaltar que existe en Internet una gran cantidad de documentación sobre el ambiente LeJOS NXJ. LeJOS NXJ se trata de un completo firmware que sustituye el oficial de LEGO.

En el momento en que se evaluaron las herramientas anteriores se optó por el lenguaje Java LeJOS NXJ, con el IDE de Eclipse ya que ambas herramientas son gratuitas. Además Java LeJOS cumple con las características de la POO (encapsulación, herencia, polimorfismo y abstracción) y permite el uso de hilos, programación de comportamientos, los cuales son muy importantes para el manejo simultáneo de sensores y motores del robot. Y se considera seguir usando el ambiente NXT-G para iniciar a los jóvenes en la programación del robot.

La Tabla 1 muestra una comparación de las principales características de dichos ambientes.

Tabla 1. Comparación de NXT-G y Java LeJOS NXJ.

Características	NXT-G	LeJOS NXJ
Versión	2.0	0.95
Lenguaje	Grafico	Java
Firmware	Standard	Propietario
IDE	Si	Plugins Eclipse y NetBeans.
Mac OSX	Si	Si
Windows	Si	Si
Linux	No	Si
Android	No	Si
Eventos	No	Si
Multihilos	Si	Si
Bluetooth Brick a PC	Si	Si
Bluetooth Brick a Brick	Si	Si
Bluetooth Brick a otros dispositivos	No	Si
Punto flotante	No	Si
Soporta dispositivos de terceros	Si	Si

El Java LeJOS NXJ es un lenguaje de programación orientado a objetos, donde se utiliza las bibliotecas tanto de Java como de NXT para la locomoción del robot y la adquisición de datos desde los sensores. En la Figura 2 se muestran las partes que componen el ambiente: tiene una barra de herramientas clásico de Eclipse [2], un área para los proyectos manejados y un área de programación, como se muestra en la figura 3.

La palabra LeJOS viene de las palabras, **Le** de “lejos” la cual es una palabra en español y **JOS** que es un acrónimo de **J**ava **O**perating **S**ystem [1]. Desarrollado por José Solórzano en el año de 1999 fue un proyecto libre que empezaría como un pasatiempo y terminaría siendo en lo que hoy se conoce como LeJOS NXJ.



LeJOS NXJ provee libre acceso a los puertos del robot, un acceso a los motores y sensores, además de contener las API's para Java.

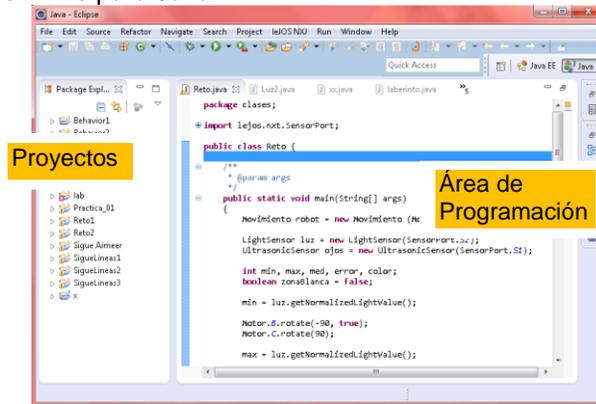


Figura 3. Entorno de desarrollo de Java LeJOS en Eclipse

3. ALGORITMOS

En este apartado se presentan los resultados obtenidos al probar los algoritmos de pista con obstáculos y el laberinto en el NXT-G y en Java LeJOS NXJ.

3.1 Pista con obstáculos

Se tiene una pista de 1.50 X 1.80 mts. de fondo blanco con una línea negra que marca el camino. Esta línea negra sube por puentes y atraviesa obstáculos (véase figura 4). El robot debe iniciar el recorrido desde la salida hasta la meta de manera autónoma subiendo puentes y evadiendo obstáculos, siempre conservándose sobre la línea negra y en el sentido que debe seguir. Los criterios que se tomaron en cuenta para determinar la eficiencia de los algoritmos fueron el tiempo de recorrido y el número de intentos exitosos. El recorrido exitoso implica que el robot inicie desde la salida y llegue a la meta de manera autónoma.

Tabla 2: Tiempo de recorrido Pista con obstáculos

Intento	Tiempo de recorrido (min)	
	Algoritmo en NXT-G	Algoritmo en Java LeJOS NXJ
1	No concluyó: pierde la línea negra.	3:27
2	5:54	3:13
3	6:16	3:15
4	No concluyó: no pudo regresar al camino después de evadir un obstáculo.	3:21
5	No concluyó: pierde la línea negra.	3:19

En la Figura 4 (a) se muestra al robot lego recorriendo la pista con obstáculos. Se observa como sube el puente y como evade una pared de piezas lego (obstáculo). La pista cuenta con dos puentes y dos obstáculos.

3.2 Laberitno

El laberinto es de 150 X 150 cm. Las paredes tienen 25 cm de alto y los pasillos tienen 25 cm. de ancho. Para este algoritmo se tomaron los mismos criterios de eficiencia que en el anterior.



En la tabla 3 se muestran el número de intentos y el tiempo que le llevó al robot recorrer el laberinto.



Figura 4. Pista con obstáculos y Laberinto.

Tabla 3: Tiempo de recorrido Laberinto

Intento	Tiempo de recorrido (min)	
	Algoritmo en NXT-G	Algoritmo en Java LeJOS NXJ
1	No concluyó.	No concluyó
2	21:23	15:47
3	No concluyó	18:22
4	No concluyó	No concluyó
5	No concluyó.	14:15

4. CONCLUSIONES

Se evaluó el desempeño de dos ambientes de desarrollo: NXT-G y Java LeJOS. El NXT-G es adecuado para personas que no tienen ninguna experiencia en el área de programación; mientras que el Java LeJOS NXJ es adecuado para estudiantes de la ISC debido a su formación en programación orientada a objetos.

Es importante mencionar que el desarrollo de los algoritmos es más fácil en Java LeJOS NXJ porque se tiene un control total sobre los sensores y los servomotores. Se encontró el inconveniente que el NXT-G solo puede manejar números enteros (positivos y negativos) y esto provoca que se vaya acumulando un error por redondeo y repercute en la locomoción del robot. En el caso de Java LeJOS como se pueden emplear variables numéricas de punto flotante el error por redondeo es mínimo por lo que el robot realiza desplazamientos más precisos.

Como pudimos observar a partir de las soluciones propuestas en los retos es mucho más extenso y complicado manejar el lenguaje de bloques NXT-G que el Java LeJOS.

BIBLIOGRAFÍA

1. Documentación y API's para Java LeJOS NXJ. <http://www.lejos.org>, consultada en agosto-diciembre de 2014.
2. Ambiente de Desarrollo Integrado Eclipse. <http://eclipse.org>, visitada en Agosto de 2014.
3. Manual Lego Mindstorms NXT 2.0, Hardware Developer Kit consultado en la página <http://www.lego.com/en-us/Default.aspx> en agosto-diciembre 2014.