

LARC 2010 – Categoría IEEE Open

Baliero Marcelo, Biganzoli Martín, Pias Gerardo,
Facultad de Ingeniería – Universidad de la República - Uruguay

Abstract— Este documento describe los diferentes pasos en la construcción de nuestro robot y su funcionamiento. Dicho proyecto nace en el marco del curso Inteligencia Artificial y Robótica de la Facultad de Ingeniería.

I. INTRODUCCIÓN

EN el marco del curso Inteligencia Artificial y Robótica, se nos planteó resolver el problema de la categoría IEEE Open, propuesto por LARC2010 [1].

Para la construcción del robot se nos proporcionó un kit robótico Lego Mindstorms® NXT® [2] y un escenario que modela el centro de distribución y los productos, cumpliendo completamente con las especificaciones planteadas por LARC.

La realización de este trabajo abarcó el período entre el 25 de Agosto y el 25 de Setiembre de 2010.

La letra completa de la tarea se encuentra disponible en http://www.fing.edu.uy/inco/cursos/robotica/laboratorio/IA_R10_laboratorio_1.pdf.

II. ANÁLISIS DE REQUERIMIENTOS

De la especificación [1] identificamos los siguientes requerimientos que el robot autónomo debía cumplir.

A. Planificación de Trayectorias

La planificación de trayectorias es un requerimiento fuerte que debíamos resolver, algunas de las preguntas que nos surgieron fueron *¿cómo se debía desplazar el robot en el escenario sin perderse?* y dada la posición inicial variable, *¿cómo y para dónde debería comenzar la navegación?*

B. Transporte

Para el caso del transporte, las preguntas que nos surgieron de forma inmediata fueron *¿cómo el robot debería tomar y transportar los cubos?* y *¿cómo debería soltar los cubos?*

C. Clasificación y Ordenamiento de Objetos

Para la clasificación, es necesario detectar los colores de las cajas, ya que no podemos asumir una disposición particular de las mismas.

El ordenamiento de objetos se debe tener en cuenta, ya

que las cajas retiradas de los fabricantes deberán ser colocadas en los vendedores de forma alineada y no deben ser abandonadas durante el trayecto.

III. SOLUCIÓN PLANTEADA

El principal desafío al que nos enfrentamos fue lograr la solución únicamente utilizando un kit robótico Lego Mindstorms® NXT®. Lo cual, si bien nos facilitaba la creación del mismo, nos limitaba en la cantidad de hardware disponible, como ser motores, sensores, pulsadores, etc. Frente a esta realidad y teniendo una idea global de la solución que buscábamos, tratamos de encontrar soluciones intermedias, incrementales, alcanzables en un corto plazo.

¿Cómo desplazarnos en el escenario sin perdernos?

Decidimos que el desplazamiento del robot fuera basado en las líneas negras existentes en el piso del escenario y que son reconocidas por el sensor del color ubicado en la parte frontal izquierda del robot. El seguimiento de las líneas permite al robot ubicarse dentro del escenario y evita que se pierda. Se utilizan también como recurso extra para determinar la ubicación del robot las dimensiones conocidas del escenario, sobre todo cuando el robot recorre grandes tramos, como el que implica el desplazamiento para el transporte de los diferentes cubos.

Dada la posición inicial variable, *¿cómo y para dónde debemos comenzar la navegación?*

Inicialmente, solamente se encuentran cajas en la zona de los fabricantes, por lo tanto, el robot comienza realizando tres giros en su posición inicial y detectando con el sensor de ultrasonido la ubicación de las diferentes pilas de cubos existentes en el área de fabricantes permitiéndole determinar hacia donde debe comenzar a moverse.

La realización de los tres giros por parte del robot depende si este detecta o no la ubicación de las cajas, si detecta inmediatamente se ubica en la dirección donde realizó el sensado y comienza a navegar hasta llegar a la línea que determina el borde del área de fabricantes.

¿Cómo tomar y transportar los cubos?

Contábamos con un único motor para lograr este fin, dado que los dos restantes se emplearon en el desplazamiento. Como estrategia frente a esta limitante

elegimos retirar el cubo ubicado en la parte inferior de la primer columna y sostener al mismo tiempo los restantes para que no se desmoronara la pila. Así el motor se destinó a mover un brazo en su parte delantera para tomar el cubo y se construyó una jaula para sostener a los otros cubos. El hecho de transportar un cubo a la vez presionado con el brazo brinda al robot una completa libertad de movimientos con la seguridad de que en ningún momento el cubo sea perdido durante el desplazamiento hacia el área de vendedores.

¿Cómo soltar los cubos?

Esta acción se vio muy simplificada por el hecho de transportar un cubo a la vez ya que simplemente implicaba que el robot se ubicara correctamente en el área de liberación, abriera el brazo con que sujeta al cubo y retrocediera para que de esa forma el cubo quede ubicado.

A. Diseño Mecánico

El diseño mecánico cuenta con tres motores (los proporcionados por el kit); dos de los mismos se utilizaron para la navegación y el restante para el manejo de la pinza que retira los cubos.

La parte frontal del robot consta de una jaula que sirve de guía y contención para la columna de los cubos. Dicha jaula cuenta con un sensor de tacto, el cual permite alinear la jaula con la columna de cubos y otro sensor también de tacto que permite determinar cuando la columna se encuentra ingresada en la jaula para poder retirar el cubo.

El sensor de color se encuentra también en la parte frontal, inclinado 45 grados para poder captar tanto la línea del piso como los colores de los cubos.

En la parte trasera del robot, colocamos el sensor ultrasónico, apuntando hacia el costado del mismo, utilizado para poder detectar los cubos cuando navegamos por las líneas.

B. Diseño o Arquitectura del Software

La arquitectura de software utilizada fue *subsumption*, implementamos distintos comportamientos los cuales al cumplirse determinadas condiciones toman o dejan el control.

Los comportamientos implementados son los siguientes:

Sigo Línea

Este comportamiento entra en acción cuando el sensor de color detecta la presencia de una línea negra debajo de él. Su comportamiento es avanzar y pierde el control cuando deja de detectar la línea.

Me Perdí

Este comportamiento toma el control cuando el robot no se encuentra sobre una línea negra y la distancia detectada por el sensor no es menor que un valor determinado (utilizado para detectar torres).

El comportamiento es girar en ambas direcciones intentando buscar la línea negra, es ahí cuando pierde el control en caso de encontrarla.

Los comportamientos antes mencionados permiten seguir líneas en el escenario.

Agarro y deajo cubo

Toma el control cuando la distancia detectada por el sensor es menor a un valor determinado, esto permite establecer cuando el robot debe realizar la maniobra de aproximación para agarrar y trasladar el cubo de madera desde el área de distribución hasta el área de vendedores, incluida la maniobra que deja el cubo.

La implementación se realizó completamente en LeJOS

C. Fortalezas

Dentro de las fortalezas de nuestra solución se encuentran la robustez y sencillez de nuestro diseño, ya que se utilizaron los recursos de la mejor forma posible logrando resultados interesantes.

IV. EXPERIMENTOS Y PRUEBAS REALIZADAS

A continuación se listan algunos videos de pruebas realizadas con nuestro robot:

<http://www.youtube.com/watch?v=XqcFOylzWcg>

<http://www.youtube.com/watch?v=EVo0LZwH1Dg>

http://www.youtube.com/watch?v=E_gDGp6iNXg&NR=1

<http://www.youtube.com/watch?v=0v8fqgGAdAg>

V. CONCLUSIONES

Contar con un kit robótico simplifica la construcción física del robot (evitando construcción a medida de piezas, soldaduras, estudio y compra de sensores, motores, etc.) sin embargo agrega complejidad debido a sus limitaciones.

Por otro lado, surgen una gran variedad de problemas físicos que se logran comprender solamente al comenzar las pruebas con el robot, como por ejemplo la deriva, problemas de calibración, de fuerza de los motores, etc.

Utilizar una arquitectura con comportamientos (subsumption) permite, si se logra una buena interacción de los mismos, un funcionamiento robusto del robot. Esto se confirma al ver el desempeño del robot sobre el escenario. Se debe tener en cuenta que la programación e implementación de los distintos comportamientos tiene un costo en tiempo considerable.

Dedicar tiempo al diseño del robot y enfocarse en la colocación de los diferentes sensores así como en la implementación del brazo nos permitió lograr una buena precisión en nuestros movimientos, destacando la importancia de lograr un control total sobre los cubos a la hora de desplazarlos por el circuito,

Teniendo en cuenta las limitaciones del kit logramos obtener un buen resultado concluyendo que un buen diseño de la solución permite sobreponerse a gran parte de los desafíos físicos que se plantearon.

REFERENCIAS

- [1] LARC 2010 IEEE Open
http://www.larc10.fei.edu.br/LARC2010_bases_libre_v10.pdf,
visitado setiembre 2010.
- [2] Sitio Web de Lego
<http://mindstorms.lego.com/>, visitado setiembre 2010
- [3] LeJos API
http://lejos.sourceforge.net/p_technologies/nxt/nxj/api/, visitado
setiembre 2010.
- [4] Blog Mindstorm NXT
<http://nxt-mindstorm.blogspot>, visitado setiembre 2010.
- [5] Electricbricks
<http://blog.electricbricks.com/>, visitado setiembre 2010.
- [6] Robin R. Murphy, "Introduction to AI Robotics", MIT Press, 2000.
- [7] Stuart Russell y Peter Norvig, "Inteligencia Artificial - Un enfoque moderno", Pearson, 2004.
- [8] García et al, "A classification of stability margins for walking robots", 5th Int. Conf. Climbing and Walking Robots, Paris.
- [9] Germán López y Santiago Margni, "Construcción de Robots a Bajo Costo", Reporte Técnico, Instituto de Computación, Facultad de Ingeniería, Universidad de la República, Montevideo, 2003.
- [10] Damián Lezama y Alexander Sklar, "Construcción de Robots Bipedos", Reporte Técnico, Instituto de Computación & Instituto de Ingeniería Eléctrica, Facultad de Ingeniería, Universidad de la República, Montevideo, 2004.
- [11] Gabriel J. Ferrer, "Using Lego Mindstorms NXT in the Classroom", Hendrix College.
- [12] Brooks, R., "A robust layered control system for a mobile robot". Robotics and Automation, IEEE Journal, 1986.