

INTRODUCCIÓN A LA ROBÓTICA Y A LAS REDES NEURONALES CON LOS VEHÍCULOS DE BRAITENBERG

INTRODUCTION TO ROBOTICS AND NEURAL NETWORKS WITH BRAITENBERG VEHICLES

Joel Ricardo Jiménez Cruz

Universidad Autónoma Metropolitana
jcjr@xanum.uam.mx

Resumen

La enseñanza de la robótica y las redes neuronales artificiales puede despertar, apoyar y motivar el desarrollo de las habilidades en las disciplinas STEAM (Ciencia, Tecnología, Ingeniería, Arte, Matemáticas) en los niños, jóvenes y no tan jóvenes. El aprovechamiento de simulaciones, artefactos electrónicos o robots físicos pueden soportar de manera muy benéfica y demostrativa estas disciplinas. La finalidad principal de este artículo es proponer el uso de los vehículos de Braitenberg como una herramienta didáctica apropiada para exponer de manera introductoria y ágil los principios tanto de la robótica reactiva como de las redes neuronales artificiales.

En este trabajo se plantea el uso de los lenguajes de programación Scratch (para educación básica y media) y Netlogo (para educación media superior y superior) para enseñar estos principios simulando los comportamientos de los vehículos. También se describe la implementación de los comportamientos en el robot físico UAMIBOT utilizando el lenguaje C. Se comentan los beneficios de estas herramientas didácticas en la enseñanza.

Palabras Claves: Netlogo, Redes Neuronales Artificiales, Robótica, Scratch, STEAM, Vehículos de Braitenberg.

Abstract

The teaching of robotics and artificial neural networks can awaken, support and motivate the development of skills in the disciplines STEAM (Science, Technology, Engineering, Art and Mathematics) in children, young and not so young.

The use of simulations, electronic devices or physical robots can support these disciplines in a very beneficial and demonstrative way. The main purpose of this article is to propose the use of Braitenberg vehicles as an appropriate didactic tool to expose in an introductory and agile way the principles of both reactive robotics and artificial neural networks. In this paper, the use of programming languages Scratch (for elementary and junior high school) and Netlogo (for high school and university education) is proposed to teach these principles simulating the behavior of the vehicles. The implementation of the behaviors in the UAMIBOT physical robot using the C language is also described. The benefits of these didactic tools in teaching are discussed.

Keywords: Artificial neuronal networks, braitenberg vehicles, netlogo, robotics, scratch, STEAM.

1. Introducción

La enseñanza de la inteligencia artificial, las redes neuronales y la robótica es muy importante para despertar y desarrollar la vocación STEAM que es un nuevo modelo dinámico de educación que añade a las disciplinas científicas (ciencia, tecnología, ingeniería, matemáticas), las humanidades (artes, diseño). Este modelo educativo es de vital importancia para llevar con éxito estudios a nivel básico, medio y superior, y para el desarrollo y crecimiento sostenido de las naciones [García, Reyes, Burgos, 2017], [Jiménez, 2018]. En este trabajo se muestran los comportamientos de robots que tienen embebido un sistema de control conductual reactivo, que por su relativa simplicidad, puede servir para explicar de manera introductoria los principios o fundamentos de la robótica y de las redes neuronales artificiales (RNA). Para lograr este objetivo, se utilizan los vehículos de Braitenberg (VB) que son criaturas reactivas que muestran comportamientos complejos e inteligentes y lo hacen por medio del uso de lo que pudieran llamarse RNA primitivas [Braitenberg, 1984]. Los conceptos de los VB son muy poderosos, simples y accesibles para los estudiantes sin conocimiento o capacitación previa y transmiten ideas fundamentales de la inteligencia artificial en general y de los sistemas de control de retroalimentación, de la robótica y de las redes neuronales, en particular.

Los robots son sistemas autónomos que habitan en el mundo real, puede sensor o registrar su medio ambiente y actúan en él para alcanzar sus metas o propósitos. Afectan su medio ambiente y son perturbados por él. En la enseñanza de la robótica se explican conceptos como autonomía, adaptación, aprendizaje, navegación y procesamiento senso-motor. La robótica educativa se ha venido utilizando en diferentes contextos; en hogares, en talleres, en colegios y de manera formal, informal, curricular y extracurricular [Ruiz, 2017].

El estudio de los robots se puede llevar a cabo desde tres enfoques principales; el deliberativo o clásico, el conductual reactivo y el híbrido. El enfoque deliberativo pretende que el robot obtenga una representación simbólica del mundo y el enfoque conductual reactivo persigue la idea biológica de que la inteligencia surge de la mediación e interrelación de los comportamientos del robot en su relación con el medio ambiente [Hossian, 2013].

Un robot reactivo es un sistema sin estados (o sin memoria) que tiene la capacidad para reaccionar a situaciones dinámicas, alivia problemas en la calibración de los sensores y efectores y evita los problemas difíciles de la representación del conocimiento en un ambiente no estructurado [Arkin, 1995]. En el lado negativo, el funcionamiento del robot puede resultar sub-óptimo; por ejemplo, un robot puede explorar una región repetidamente sin aventurarse en nuevos territorios, puesto que no tiene memoria de que ya visitó esa región con anterioridad.

Las RNA se pueden utilizar para la generación de comportamientos en los robots, asignando diferentes conexiones y diferentes pesos, con el fin de obtener diferentes comportamientos. En este caso, los comportamientos que tienen pesos más grandes en sus conexiones tienden a relegar o subyugar a los comportamientos que tienen pesos más pequeños. Por lo tanto, los comportamientos podrían priorizarse e integrarse para desarrollar las tareas encomendadas [Wyeth, 1997].

La enseñanza y explicación de los principios de las RNA y la robótica inteligente se puede impartir a niños, jóvenes y adultos utilizando simulaciones apropiadas o robots físicos para cada edad o situación educativa en la que se desee incursionar. En nuestro caso, hemos utilizado a nivel universitario, los lenguajes de programación Scratch y Netlogo y aplicaciones reales en C con el robot físico

UAMIBOT; las simulaciones en Scratch las hemos utilizado en la materia de Fundamentos de la Programación y las simulaciones en Netlogo y la implementación en el robot físico se han empleado en la materia de Inteligencia Artificial.

Por otro lado, para la enseñanza de la programación de simulaciones o de robots se puede utilizar el lenguaje Scratch cuando se trata de niños y adultos no versados, el lenguaje Netlogo en el caso de jóvenes de educación media y C o Python cuando se trata de jóvenes o adultos de educación media o superior. Los robots pueden ser robots físicos comerciales (como Thymio II), robots caseros, kits educacionales (como LEGO) o desarrollos propios electrónicos analógicos o digitales.

En las secciones que siguen, se describe la metodología utilizada, los resultados obtenidos a partir de las simulaciones y experimentos realizados, se discute su uso en la enseñanza y se concluye con las experiencias obtenidas en este trabajo.

2. Métodos

En esta sección se propone mostrar cómo llevar a cabo la enseñanza de los principios de la robótica y las RNA utilizando los fundamentos conceptuales de los VB y la aplicación en simulaciones o en robots físicos.

Con el uso de los lenguajes de programación Scratch (para educación básica y media) y Netlogo (para educación media y superior) es posible enseñar cómo realizar simulaciones de los comportamientos de los vehículos V2 y V3. El lenguaje C se ha utilizado para programar estos comportamientos en el robot físico UAMIBOT.

Los vehículos de Braitenberg

Los VB son abstracciones y concepciones mentales de vehículos o criaturas que surgieron de los experimentos que realizó el neuroanatomista Valentino Braitenberg en su laboratorio del instituto Max Plank en Alemania en los años ochenta. Su trabajo seminal ha tenido un impacto muy importante en varias áreas del conocimiento relacionadas con la robótica, en lo referente a su concepción, diseño y construcción [Braitenberg, 1984].

Los VB describen una serie de experimentos conceptuales que demuestran cómo el comportamiento inteligente en vehículos o robots puede emerger del acoplamiento senso-motor. Los conceptos básicos de los VB se han utilizado en diversos estudios de robótica y de sistemas multiagentes, tanto en experimentos físicos como en simulados. Por ejemplo, se han utilizado para:

- Localización de una fuente de olor [Lilienthal, Duckett, 2003], [Mamduh et al, 2014], [Takei et al, 2014], [Kamarudin et al, 2014].
- Cobertura de una área para aplicaciones que van de la limpieza en pisos hasta el desminado [Rano, Santos, 2017].
- Cuestiones éticas en dispositivos artificiales sobre la autonomía y la moralidad como el egoísmo y el altruismo, [Headleand, Teahan, 2016].
- Búsqueda [Saavedra, Gutiérrez, Enríquez, 2016].
- Orientación de un pez en un flujo uniforme contracorriente [Salumae, et al, 2012].

La mayoría de los ejemplos de investigación en esta área se ha centrado en los vehículos tipos 2 y 3, por su simplicidad y comportamiento bien definido [Rano, 2012].

En la figura 1 se muestran los comportamientos de los VB; V2 y V3 que se utilizaron en este trabajo. Los comportamientos que pueden surgir de acuerdo al tipo de conexiones (cruzadas o laterales) y pesos (excitatorios o inhibitorios) en sus conexiones son: cobarde o miedoso (V2a), agresivo o de odio (V2b), amoroso (V3a) y explorador (V3b).

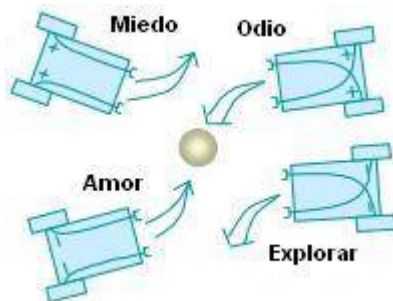


Figura 1 Comportamientos surgidos en los VB debido al tipo de conexiones y pesos.

Las redes neuronales artificiales

En las RNA se realiza una transformación (lineal o no lineal) de la entrada x_i para obtener una respuesta y_i de la forma $y_i = g(x_i)$. A la red se le presentan patrones tanto a la entrada y a la salida con el fin de que la red aprenda esta transformación, es decir, se entrena a la red para que reconozca esos patrones de entrada y produzca la salida deseada. La transformación consiste en ajustar los valores o pesos que tienen las conexiones entre las diferentes unidades, nodos o neuronas. El entrenamiento se puede realizar, para redes sencillas y pequeñas, en forma manual y para redes más complejas y grandes se utilizan algoritmos de entrenamiento como el de retropropagación [Ruan, Dai, 2012].

Existen varios tipos de RNA, principalmente [Kröse, Van der Smagt, 1999]:

- Redes supervisadas con aprendizaje dirigido y no supervisadas con aprendizaje autodirigido.
- Redes hacia adelante y con retroalimentación y autoretroalimentación

En las redes primitivas de los VB se tienen redes de una capa con una sola entrada, una sola salida y una función de activación representada por una constante que puede ser positiva o negativa. A su vez, la conectividad que se da entre sus unidades puede ser lateral o cruzada.

En la figura 2 se ilustran las posibles conexiones que se han programado en los diferentes lenguajes para que emerjan los diferentes comportamientos de los VB. En esta figura, S_L y S_R son los sensores, M_L y M_R son los motores (actuadores) y $C1$, $C2$, $S1$ y $S2$ son los pesos de las conexiones. La velocidad del motor está dada por la combinación lineal de la entrada registrada y los pesos entre sus conexiones, que se expresa en las ecuaciones 1 y 2.

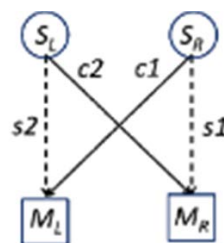


Figura 2 Las posibles conexiones y pesos en las redes de los VB.

$$M_L = C_1 S_R + S_2 S_L \quad (1)$$

$$M_R = C_2 S_L + S_1 S_R \quad (2)$$

Esta conectividad simple es la que produce los comportamientos en los VB y es el principio de la programación de RNAs más complejas. En estas redes simples se observa que los ajustes en los pesos (entrenamiento) es la base para explicar y entender de manera intuitiva cómo se lleva a cabo el aprendizaje en las redes artificiales y en las redes biológicas que están presentes en las máquinas y en los seres vivos, ya sean estas plantas, organismos pluricelulares o seres humanos.

Hay una relación entre las conexiones sensor-actuador y los comportamientos de un robot. Los VB no tienen un proceso explícito de toma de decisiones, el comportamiento del robot depende únicamente de la conexión entre sensores y actuadores.

Aplicando y extendiendo las ideas de las RNAs primitivas de Braitenberg, se puede incursionar en otro tipo de redes neurales más sofisticadas con aprendizaje supervisado que se pudieran utilizar para entrenar a robots que realicen otras tareas. Las redes más sencillas como las de McCulloch y Pitts y las de Rosenblatt fueron criticadas por sus limitaciones e incapacidad para realizar ciertos cálculos computacionales como la clasificación no lineal, pero en realidad se podrían utilizar en los robots con un entrenamiento manual.

Por ejemplo, Wyeth [1997] y Browning, Wyeth [1998] realizaron una mejora en los comportamientos de los VB utilizando las redes de una sola capa Adaline y el Perceptrón. La diferencia en estas redes radica en la función de transferencia de la neurona; en la primera red, las unidades de procesamiento son lineales y la segunda red son no lineales.

En la literatura hay varios estudios que demuestran los beneficios del uso de la programación de agentes o robots simulados o reales como métodos más efectivos, atractivos, motivadores y entretenidos en comparación con la educación sin el uso de ellos. La interacción con estas ayudas didácticas aumentan la motivación, el compromiso y la actitud hacia la educación. Se cree que la enseñanza de la programación desarrolla en los estudiantes habilidades del pensamiento creativo,

de la algorítmica, de la capacidad de abstracción y del pensamiento computacional que involucran el procesamiento de la información y que contribuyen a mejorar las habilidades en la resolución de problemas [Karim, Lemaignan, Mondada, 2015], [López, Andrade, 2013]. Por ejemplo, Çankaya, Durak y Yünkül [2017] evaluaron las opiniones de los estudiantes de bachillerato que aprendieron a programar robots, aplicando una prueba de resolución de problemas similar a la de PISA 2012, una prueba de evaluación de rendimiento al programar y una entrevista. Encontraron una relación significativa positiva entre las habilidades de resolución de problemas de los estudiantes y sus puntajes de rendimiento al programar.

En la sección que sigue se muestran las simulaciones en los lenguajes Scratch y Netlogo y el comportamiento del robot físico UAMIBOT que en su conjunto ejemplifican el uso de los VB como una herramienta didáctica apropiada para exponer de manera introductoria y ágil los principios de la robótica reactiva y de las RNA.

3. Resultados

En las simulaciones y experimentos realizados se hacen una serie de supuestos que simplifican la dinámica de los vehículos. En primer lugar, la velocidad de cada rueda se controla linealmente por el valor que se alimenta al motor; no se da importancia a los efectos de inercia o fricción, ni al comportamiento no lineal del efector. En la simulación, se emula la cinemática del vehículo, no su dinámica. Las colisiones se evitan y no afectan al movimiento del robot. En segundo lugar, los sensores devuelven un valor normalizado ideal que representa la proximidad del objeto al sensor. En los programas de simulación realizados se pueden efectuar cambios en los parámetros de varias entidades, como:

- Medio ambiente- en particular el número y la posición de los obstáculos y objetivos,
- Controlador- incluye el tipo de unidades de activación y la conectividad neural utilizada para cierto comportamiento, y
- Robot- permite la actuación de las ruedas motrices y la posición y alcance de los sensores.

Simulaciones en Netlogo

NetLogo es un lenguaje de programación y es un medio para simular principalmente fenómenos sociales, ingenieriles y naturales. Se ha diseñado para usarse en la educación y en la investigación. Es adecuado particularmente para modelar, en forma concurrente, sistemas dinámicos que evolucionan en el tiempo. Esto hace posible explorar las conexiones existentes entre las conductas de los agentes y los patrones que emergen a partir de sus interacciones bajo diversas variables y condiciones [Tisue, Wilensky, 2004].

Algunos de los temas que se han estudiado y simulado en Netlogo incluyen a los autómatas celulares, la teoría de juegos, la computación evolutiva, los sistemas de control con retroalimentación positiva y negativa, la dinámica de poblaciones [Pérez, Jiménez, 2013], la optimización, la autoorganización, la vida artificial, la inteligencia artificial, la evolución, las redes, los mercados, el caos. Los modelos comparten el tema de los sistemas complejos y de los fenómenos emergentes.

En este trabajo, sugerimos que Netlogo se puede utilizar convenientemente para enseñar a programar las simulaciones de los comportamientos cobarde, agresivo, amoroso y explorador de los VB.

Un extracto del programa utilizado en las simulaciones se muestra en la tabla 1.

En las simulaciones se tiene como objeto de búsqueda una luz, pero bien podría tratarse de una sustancia química, un alimento, etc.

Tabla 1 Códigos correspondientes a la definición de los comportamientos de los VB.

<pre>if conducta = "covarde" [set motor-izq (pesoW * sensor-izq); set motor-der (pesoW * sensor-der);]</pre>	<pre>if conducta = "agresiva" [set motor-izq (pesoW * sensor-der); set motor-der (pesoW * sensor-izq);]</pre>	V2
<pre>if conducta = "amorosa" [set motor-izq (0.6 - pesoW * sensor-izq); set motor-der (0.6 - pesoW * sensor-der);]</pre>	<pre>if conducta = "exploradora" [set motor-izq (0.6 - pesoW * sensor-der); set motor-der (0.6 - pesoW * sensor-izq);]</pre>	V3

En la figura 3 se presenta la simulación realizada en Netlogo que muestra como el vehículo V2a cobarde o temeroso se acerca y se aleja de la fuente de luz, mientras que el vehículo agresivo V2b irrumpe sobre la fuente de luz. En estas simulaciones se puede apreciar la respuesta autónoma del vehículo en base al procesamiento senso-motor que realiza y a la interrelación con su medio ambiente.

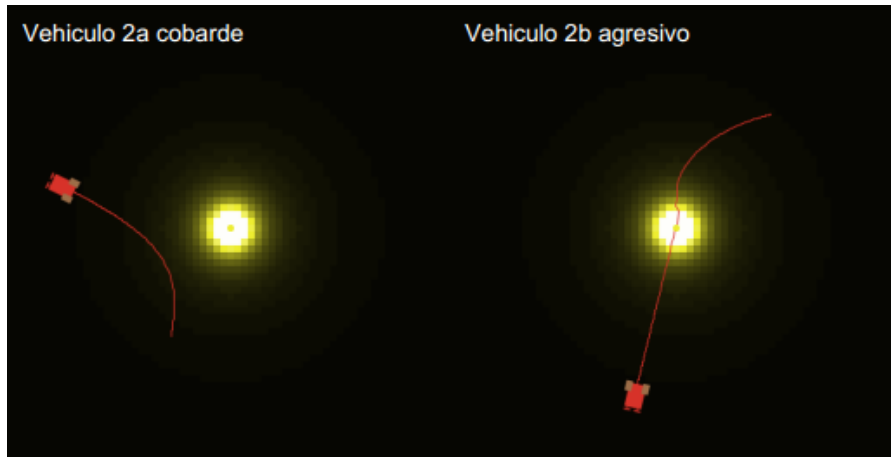


Figura 3 Comportamiento del vehículo V2.

En la figura 4 se pueden observar las conductas amorosas del vehículo V3a que se detiene frente a la luz y las conductas exploradoras del vehículo V3b. Estas conductas ejemplifican los conceptos de navegación y búsqueda que en un momento determinado se desean implementar de manera autónoma e inteligente en aplicaciones de robots en ambientes físicos o virtuales en Internet.

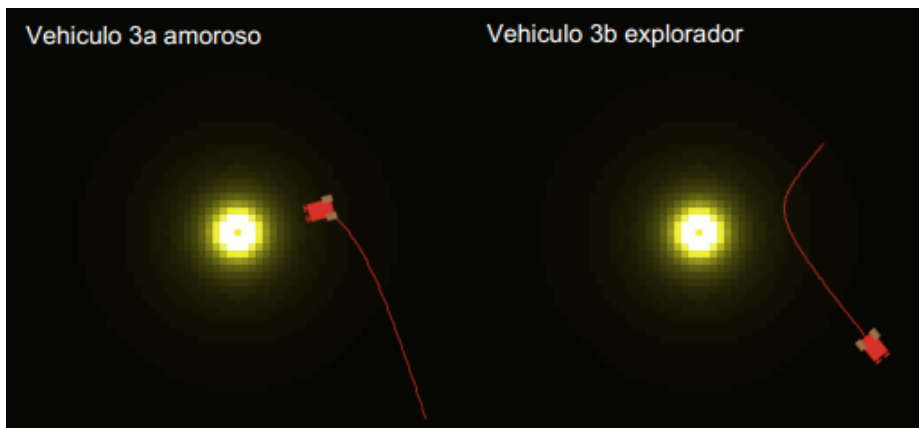


Figura 4 Corridas en Netlogo del vehículo V3.

Experimentos de los VB con el robot UAMIBOT

El robot móvil UAMIBOT tiene varios sensores y dos motores. Entre los sensores se encuentran dos fotosensores LDR. En el microcontrolador PIC, se descargaron, con el apoyo del Compilador C CSS versión 4.130, los programas de los VB que se codificaron en el lenguaje C, [Jiménez, Cruz, 2013].

En los programas se incluyen librerías generales para la tarjeta de control y librerías particulares del robot para la lectura de los sensores y la activación de los motores. Una parte de código en C para el robot UAMIBOT con la cual se implementa el vehículo amoroso V3a (a mayor intensidad de luz, el robot se va deteniendo y a menor intensidad de luz, el robot va avanzando a mayor velocidad):

```
while (true) {                               // Comienza ciclo infinito
/* Se activa la entrada analógica RA5 para medir la intensidad de luz
  de la fotocelda Izquierda*/
set_adc_channelv(5);
delay_us(20);                               // Retraso para estabilidad del canal analógico
LDR5 = read_adcv(); // Se lee el canal analógico
set_adc_channelv(6); // Se activa la entrada de la fotocelda RA6 Derecha
delay_us(20);                               // Estabilización del canal analógico
LDR6 = read_adc(); // Lectura de la conversión AD del canal analógico derecho
Avanzar(LDR5/20,LDR6/20); /*Rutina creada en librería para avanzar en función de
  la luz (1/20: factor de conexión entre sensor (fotoresistencia) y actuador (motor)*/
delay_ms(300);                               // El robot se pone en movimiento
}                                             // Fin del ciclo
```

En la figura 5 se muestran fotografías del comportamiento del vehículo V3a programadas en el lenguaje C en el robot UAMIBOT. Con la ayuda de un celular se muestra la respuesta reactiva del robot ante el estímulo luminoso. Físicamente se siente como si el robot respondiera de manera inteligente ante la situación imperante en el medio ambiente.

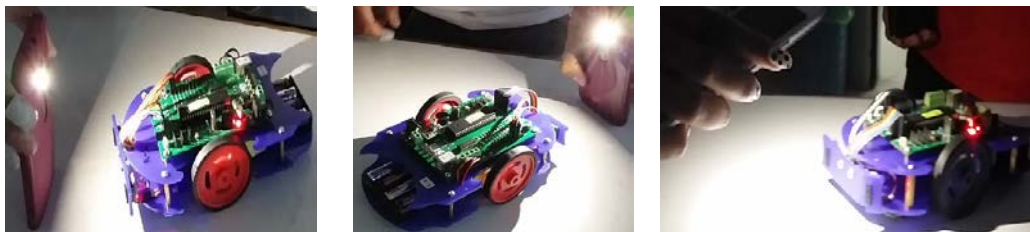


Figura 5 Fotos que ejemplifican el comportamiento amoroso del vehículo V3a.

Simulación con el lenguaje de programación Scratch

Scratch es un lenguaje de programación apropiado para niños o personas que empiezan a programar por primera vez. Cuenta con lo necesario para el desarrollo de juegos, historias interactivas, arte gráfico, animación y proyectos multimedia en general. Los programas se crean ensamblando bloques en un entorno gráfico que incluyen una gran colección de imágenes, sonidos y una herramienta (parecida a Paint) para crear dibujos, producir música y editar efectos de sonido.

Los programas en Scratch se construyen a partir de bloques gráficos que se van encajando uno a continuación del otro. Estos bloques son como piezas de rompecabezas por que la forma en que se acomodan y que hacen cierto sentido, previniendo de hacer combinaciones inválidas. De esta manera se mantiene una sintaxis de programación apropiada para que los nuevos programadores aprendan la forma correcta de ensamblar y formular la lógica de programación. Con scratch se logran aprender principios y técnicas de programación que se pueden aplicar posteriormente en otros lenguajes de programación.

Scratch cuenta con 8 bloques de programación; 3 bloques están destinados a las estructuras generales de programación (Operadores, Eventos, Control), 4 bloques se utilizan para la programación de multimedia (Movimiento, Apariencia, Sonido, Lápiz) y un bloque para la utilización de Sensores.

Como un ejemplo de aplicación en el lenguaje Scratch se implementó el VB amoroso.

En la figura 6 se expone el comportamiento del vehículo amoroso simulado en Scratch. La interfaz con el usuario se divide principalmente en tres partes o columnas; en la primera se observa el funcionamiento del programa con los objetos o sprites creados; en la segunda, los bloques disponibles para realizar un programa; y en la tercera columna se aprecia el script correspondiente al objeto luz.

En el programa se codificó que la luz se quedara en un lugar por 5 segundos y luego cambiara de lugar aleatoriamente. El objeto nombrado Vehículo también tiene asociado un código que le permite dirigirse hacia la luz. En particular, en esta simulación se puede apreciar la navegación y el procesamiento senso-motor que tiene el robot en función de la relación con su medio ambiente.

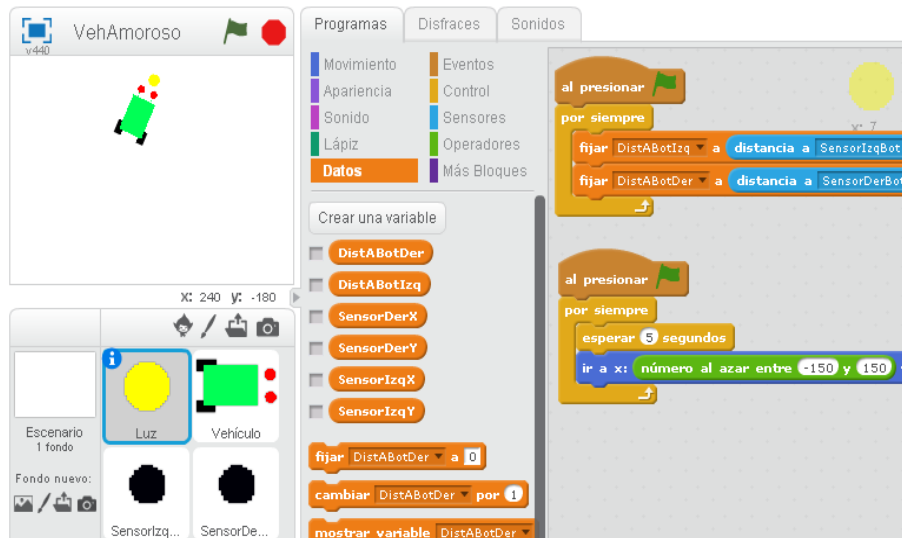


Figura 6 Implementación del VB amoroso en Scratch.

Los principios teóricos de las RNA y de la robótica, así como las simulaciones y los experimentos que en este trabajo reportamos, los hemos utilizados en clases a nivel universitario; las simulaciones en Scratch las hemos empleado en la materia de Fundamentos de la Programación y las simulaciones en NetLogo y el uso de C en el robot físico se han implementado en la materia de Inteligencia Artificial. En cualquiera de los casos se procede inicialmente a familiarizarse con la teoría tanto en clase, como en las tareas que se dejan, a continuación se proporcionan fragmentos de código para que los alumnos implementen las partes que faltan para completar el experimento.

En la entrega de los reportes correspondientes y en la observación de su avance académico en el laboratorio, se ha constatado que los alumnos comprenden y asimilan mejor el material teórico al realizar los experimentos prácticos.

4. Discusión

La enseñanza de los VB y su programación con RNA primitivas puede utilizarse para mostrar otros temas avanzados. Por ejemplo los comportamientos de los VB pueden usarse para simular la cognición, debido a que su comportamiento se puede describir y estudiar en términos psicológicos. Como se ha visto en los experimentos realizados; ligeros cambios en las conexiones de las redes de estos vehículos

producen cambios significativos en su comportamiento. Esta idea se puede generalizar para comprender como funcionan la plasticidad y el aprendizaje cerebral.

Los comportamientos de los VB están relacionados con sus mecanismos internos que son deterministas, pero las interpretaciones de un observador externo pueden ser diferentes a los mecanismos programados. Estos comportamientos aparentemente sofisticados pueden surgir a partir de elementos muy simples y de la interacción del agente con su entorno. Por otro lado, hay que tomar en cuenta que las señales que se obtienen de los sensores en el mundo real son a menudo muy ruidosas y correlacionadas sólo hasta cierto punto con las variables que se desean medir [Pfeifer, 1997].

La autonomía que se observa en las simulaciones puede interpretarse como la autosuficiencia de un agente cuando es capaz de sostenerse a sí mismo por un periodo largo de tiempo (por ejemplo, buscando energía cuando ésta se encuentre baja). La autonomía depende también de saber elegir la acción correcta de acuerdo a las circunstancias del medio ambiente y del estado interno del agente.

Los robots autónomos enseñan ideas importantes sobre la inteligencia, como, por ejemplo, que no se requiere de un control central para realizar las tareas encomendadas. En lugar de un control central se producen interacciones entre los controles locales y emerge la auto-organización [Pfeifer, 1997].

Cada tipo de VB puede presentar un comportamiento más complejo en la medida que aumenta la complejidad del entorno. Por ejemplo, la adición de varias fuentes de estimulación modificaría la navegación y la ejecución de las tareas del agente en su medio ambiente. Se puede observar en los VB que el comportamiento adaptable al entorno surge sin necesidad de procesar una gran cantidad de información. Para un observador puede parecer inteligente, pero el vehículo está simplemente reaccionando a su ecosistema, aunque éste sea dinámico e impredecible. En conclusión, se puede decir que un comportamiento complejo puede emerger de mecanismos relativamente simples al interactuar con el medio ambiente.

En otro orden de ideas y en nuestra experiencia, el empleo de las simulaciones y el robot nos han ayudado a darnos cuenta que las interacciones profesor-alumno y

alumno-instrumentos didácticos son fundamentales para lograr una mejor visualización y comprensión de los conceptos teóricos, abstractos y prácticos y para mejorar las experiencias de aprendizaje tanto individuales como colaborativas. Los docentes necesitamos sentirnos cómodos con estos apoyos. Tal comodidad puede lograrse con la selección de la tecnología apropiada, una capacitación adecuada y una participación activa/proactiva.

5. Conclusiones

En la experiencia en clase nos hemos percatado que la exposición de los temas de las RNA y la robótica no es tan sencilla como pareciera. Sin embargo, las simulaciones que se implementaron en Netlogo y Scratch y los experimentos con el robot físico UAMIBOT nos han ayudado para mostrar, integrar, explicar y enseñar estos temas de manera armoniosa y entendible. A los estudiantes les pareció una forma de aprendizaje interesante y hasta cierto punto divertida.

Los lenguajes y plataformas que se han mostrado en este trabajo, pueden apoyar, motivar y ayudar a desarrollar las habilidades en las áreas de STEAM en los niños, jóvenes y no tan jóvenes. Pueden ser de mucha utilidad para introducir los conceptos fundamentales de la robótica inteligente como navegación, evitación de obstáculos, emergencia de la inteligencia, autonomía, procesamiento senso-motor e interrelación con el medio ambiente.

Los conceptos de los VB son muy poderosos, simples y accesibles para los estudiantes sin conocimiento o capacitación previa y transmiten ideas fundamentales de los sistemas de control de retroalimentación, de la robótica, de las redes neuronales y de la inteligencia artificial. Estos principios básicos de sistemas de retroalimentación permiten entender de mejor manera, la interacción de agente o robot con su medio ambiente [Stolkin, Sheryll, Hotaling, 2007].

La introducción de estos temas permite abstraer conceptos importantes que pueden incentivar el diseño e implementación posterior de diferentes aplicaciones robóticas. Por ejemplo, se puede proponer la programación de extensiones o cambios a la arquitectura de Braitenberg que incluya capacidades de selección de acción cuando se involucran más comportamientos [Lambrinos, Scheier, 1995].

La adaptación es también un tema avanzado que puede responder a algunas preguntas de cómo el robot hace frente a las situaciones del mundo real, cómo las aprovecha, cómo explota la física, cómo se beneficia de la dinámica del mundo real, etc., que se podrían incursionar, responder y enseñar en base a los conceptos desarrollados en este artículo.

6. Bibliografía y Referencias

- [1] Arkin, R.C. (1995). Reactive Robotic Systems. Article in Handbook of Brain Theory and Neural Networks, ed. M. Arbib, MIT Press, pp. 793-796. <http://www.cc.gatech.edu/ai/robot-lab/online-publications/handbook.pdf>
- [2] Braitenberg, V. (1984). Vehicles: Experiments in Synthetic Psychology. MIT Press. <http://math.hws.edu/vaughn/cpsc/336/docs/vehicles.pdf>
- [3] Browning, B., Wyeth, G. (1998). Neural Systems for Integrating Robot Behaviours. Australian Conference on Neural Networks. <https://pdfs.semanticscholar.org/f153/689af28007a04cd19bf24e9dd7fe777a51b1.pdf>
- [4] Çankaya, S., Durak, G., Yünkül, E. (2017). Education on Programming with Robots: Examining Students' Experiences and Views. Turkish Online Journal of Qualitative Inquiry (TOJQI). Volume 8, Issue 4, October: 428-445. DOI: 10.17569/tojqi.343218. <http://dergipark.gov.tr/download/article-file/357017>
- [5] García, Y., Reyes, D., Burgos, F. (2017). Actividades STEM en la formación inicial de profesores. Nuevos enfoques didácticos para los desafíos del siglo XXI. Diálogos educativos, Vol. 17, Nº. 33. <http://revistas.umce.cl/index.php/dialogoseducativos/article/view/1168>.
- [6] Headleand, C., Teahan, W. (2016). Towards ethical robots: Revisiting Braitenberg's vehicles. SAI Computing Conference. <http://ieeexplore.ieee.org/abstract/document/7556023/>.
- [7] Hossian, A., Monte, G., Olivera, V. (2013). Análisis del Comportamiento de Robots Móviles con RNA. Un Acercamiento desde el Paradigma Reactivo. Revista Latinoamericana de Ingeniería de Software, 1(1): 18-24. <http://sistemas.unla.edu.ar/sistemas/redisla/ReLAIS/relais-v1-n1-p-18-24.pdf>

- [8] Jiménez, J. (2018). STEAM y las culturas Making, Tinkering y Do it Yourself. Memorias del XXXI Simposio Internacional de TIC en la educación, SOMECE. <http://www.telematica.ccadet.unam.mx/someceweb/#/reader/chapter/874>.
- [9] Jiménez, J., Cruz, D. (2013). UAMIBOT: un robot móvil didáctico multifunciones. *Avances de Ingeniería Electrónica*. SENIE 2013. <http://sgpwe.izt.uam.mx/files/users/uami/jcjr/UamiBotSENIE13.pdf>
- [10] Kamarudin, S.M., K., Shakaff, A.Y.M., Zakaria, A., Abdulla A.H. (2014). Comparison of Braitenberg Vehicles with Bio-Inspired Algorithms for Odor Tracking in Laminar Flow. *Australian Journal of Basic and Applied Sciences*, 8(4), Pages: 6-15. https://www.researchgate.net/profile/Syed_Muhammad_Mamduh_Syed_Zakaria/publication/261672644_Comparison_of_Braitenberg_Vehicles_with_Bio-Inspired_Algorithms_for_Odor_Tracking_in_Laminar_Flow/links/0a85e534f6ca0d99fb000000.pdf.
- [11] Karim, M., Lemaignan, S., Mondada, F. (2015). A review: Can robots reshape K-12 STEM education?, *IEEE International Workshop on Advanced Robotics and its Social Impacts (ARSO)*, pp. 1-8. doi: 10.1109/ARSO.2015.7428217. https://infoscience.epfl.ch/record/209219/files/2015_ehsan_CanRobotsReshapeStemEducation.pdf
- [12] Kröse, B., Van der Smagt, P. (1999). *An Introduction to Neural Networks*, University of Amsterdam, Faculty of Mathematics & Computer Science, The Netherlands. Eighth Edition. <https://www.infor.uva.es/~teodoro/neuro-intro.pdf>
- [13] Lambrinos, D., Scheier, Ch. (1995). *Extended Braitenberg Architectures*. Technical Report AI Lab no. 95.10, Computer Science Department, University of Zurich. <http://www8.cs.umu.se/kurser/TDBD17/VT06/utdelat/Assignment%20Papers/Extended%20Braitenberg%20Architectures.pdf>.
- [14] López, P., Andrade, H. (2013). Aprendizaje con robótica, algunas experiencias. *Revista Educación* 37 (1). <http://www.redalyc.org/articulo.oa?id=44028564003>
- [15] Rano, I., Santos, J. (2017). A biologically inspired controller to solve the coverage problem in robotics, *Bioinspiration & Biomimetics*. <http://iopscience.iop.org/article/10.1088/1748-3190/aa714c/pdf>.

- [16] Lilienthal, A., and Duckett, T (2003). "Experimental analysis of smelling braitenberg vehicles," In Proceedings of the IEEE International Conference on Advanced Robotics (ICAR 2003), pages 375–380, Coimbra, Portugal. www.ra.cs.uni-tuebingen.de/publikationen/2003/lilien03icar2003_col.pdf.
- [17] Pérez, J., Jiménez, J. (2013). Modelado y simulación preliminar del asentamiento poblacional en una entidad del estado de México. *Avances de Ingeniería Electrónica. SENIE 2013*, páginas 452 – 463. <http://sgpwe.izt.uam.mx/pages/cbi/jcjr/archivos/ModeloNetLogoSENIE13.pdf>
- [18] Pfeifer, R. (1997). Teaching powerful ideas with autonomous mobile robots. *Journal of Computer Science Education* 7: 161-186. <http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=AD5D837752AD262DBBEDCB56A84FADF3?doi=10.1.1.55.8798&rep=rep1&type=pdf>.
- [19] Rano, I. (2012). A Model and Formal Analysis of Braitenberg Vehicles 2 and 3. *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 910-915. <http://www.neuroinformatik.ruhr-uni-bochum.de/data/documents/rano12model.pdf>.
- [20] Ruan, X., Dai, L. (2012). Vehicle Study with Neural Networks. *International Conference on Solid State Devices and Materials Science. Physics Procedia* 25: 814-821. doi: 10.1016/j.phpro.2012.03.162. <https://core.ac.uk/download/pdf/82017621.pdf>.
- [21] Ruiz, F. (2017). Diseño de proyectos STEAM a partir del currículum actual de Educación Primaria utilizando Aprendizaje Basado en Problemas, Aprendizaje Cooperativo, Flipped Classroom y Robótica Educativa. Tesis doctoral, Departamento de Ciencias de la Educación, Universidad CEU Cardenal Herrera de Valencia, España. <http://dspace.ceu.es/handle/10637/8739>
- [22] Saavedra, M., Gutiérrez, M., Enríquez, L (2016). Análisis de comportamientos de vehículos de Braitenberg para búsqueda robótica usando el robot lego EV3. *Fides Et Radio*, Vol 12, páginas 155-166. <http://ucbconocimiento.ucbcbba.edu.bo/index.php/FER/article/view/1193>.
- [23] Salumae, T., Raño, I., Akanyeti, O., Kruusmaa, M. (2012). Against the flow: A Braitenberg controller for a fish robot. *IEEE International Conference on*

- Robotics and Automation. https://www.researchgate.net/publication/254041232_Against_the_flow_A_Braitenberg_controller_for_a_fish_robot.
- [24] Stolkin, R., Sheryll, R., Hotaling, L. (2007). Braitenbergian experiments with simple aquatic robots, OCEANS, Vancouver, BC, pp. 1-7. doi: 10.1109/OCEANS.2007.4449202. <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4449202&isnumber=4449115>.
- [25] Takei, Y., Shimizu, Y., Hirasawa, K., Nanto, H. (2014). Braitenberg's vehicle-like odor plume tracking robot. SENSORS, IEEE. <http://ieeexplore.ieee.org/abstract/document/6985243/>.
- [26] Tisue, S., Wilensky, U. (2004). NetLogo: A Simple Environment for Modeling Complexity. International Conference on Complex Systems, Boston. <http://ccl.northwestern.edu/papers/netlogoiccs2004.pdf>
- [27] Wyeth, G. (1997). Neural Mechanisms for Training Autonomous Robots. Proceedings of Mechatronics and Machine Vision in Practice Conference, publ. IEEE Computer. <https://pdfs.semanticscholar.org/7464/d409eda33205185322392de079e08d30a800.pdf>.