

LECCIÓN AVANZADA DE PROGRAMACIÓN EV3

Control Proporcional



Por Droids Robotics
Código por The Construction Mavericks



Objetivos de la lección

1. Aprender qué significa el control proporcional y porqué usarlo.
2. Aprender a aplicar el control proporcional al sensor Giroscópico, de Color i Ultrasónico.

Prerrequisitos: Bloques matemáticos, Calibración del sensor de color, Cables de datos

Aprender y discutir el Control Proporcional

- En nuestro equipo, entendemos “proporcional” como un juego.
- Poned una venda en los ojos a un miembro del equipo. Él o ella tiene que desplazarse tan rápido como pueda por una habitación y detenerse exactamente en una línea dibujada en el suelo (utilizad cinta adhesiva para dibujar la línea).
- El resto del equipo tiene que darle las instrucciones (comandos).
- Cuando vuestro compañero esté lejos, debe moverse rápido y con pasos grandes. Pero, a medida que se acerca a la línea, si continua corriendo, se la pasará. Así que, para que eso no suceda, debéis guiar al compañero para que vaya más lento y con pasos más pequeños.
- Debéis programar el robot del mismo modo!



¿Por qué el Control Proporcional?

- ¿Qué significa *proporcional*?
 - El robot se mueve proporcionalmente – movimiento más o menos rápido según se acerca a su objetivo.
 - En un seguidor de línea, el robot puede hacer un giro más fuerte si está más lejos de la línea.
- El Control proporcional permite ser más preciso y más rápido.
- El pseudocódigo para cada programa de Control Proporcional consta de dos etapas:
 1. **Calcular el error** → distancia del robot al objetivo
 2. **Hacer la corrección** → hacer que el robot tome una corrección que es proporcional al error (por eso se llama control proporcional). Se debe multiplicar el error por un factor de escala para determinar la corrección.

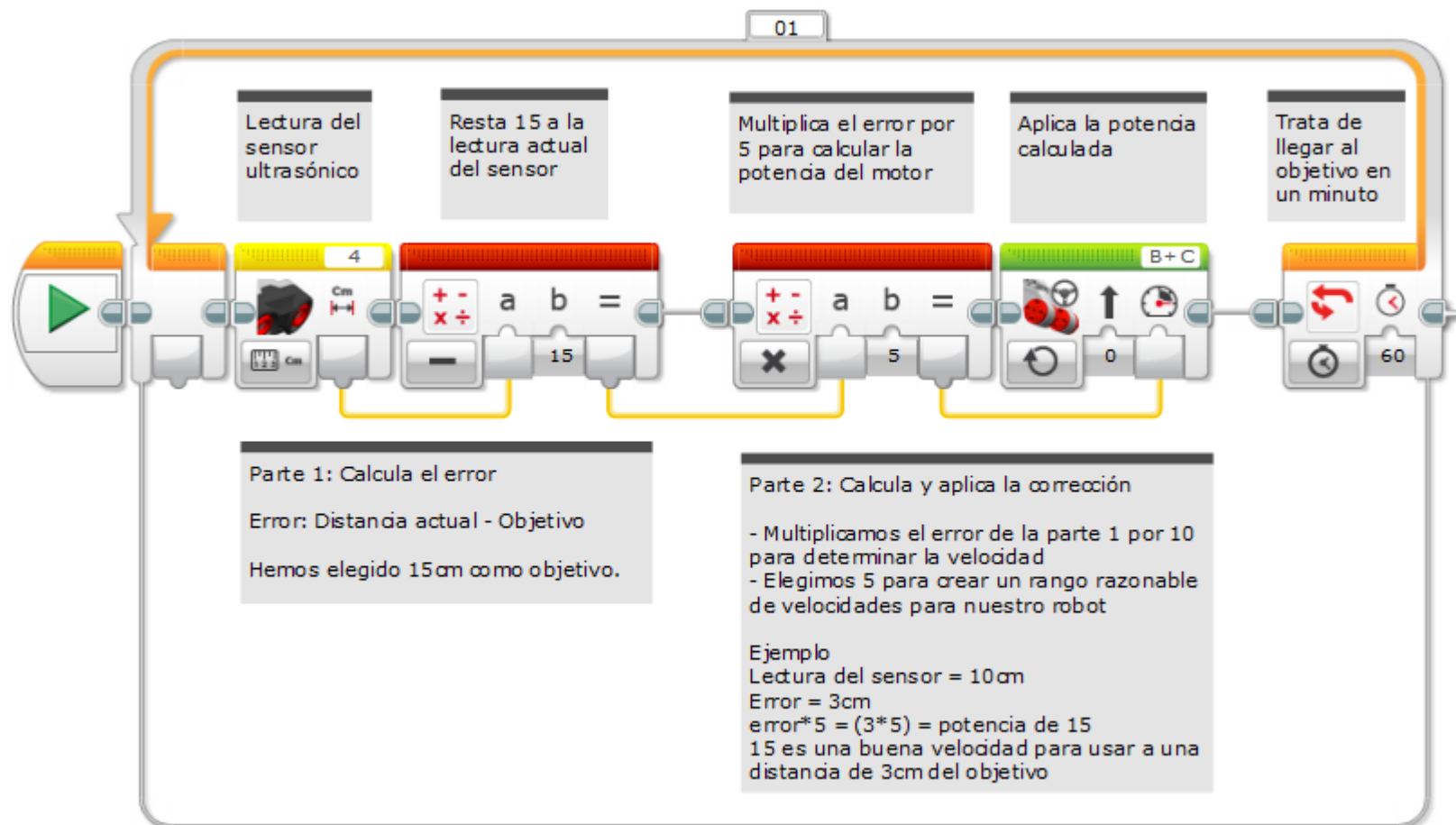
Desafíos

- Para aprender a utilizar el control proporcional, proponemos tres retos diferentes :
 - Perro Seguidor: Uso del control proporcional con el sensor ultrasónico para conseguir que el robot se quede en todo momento a 15 cm de distancia de la persona (incluso cuando ésta se mueve)
 - Seguidor de línea: Uso del control proporcional con el sensor de luz para conseguir que el robot siga la línea suavemente. (Más detalles en la lección del Seguidor de Línea Proporcional)
 - Sensor giroscópico: Uso del control proporcional con el sensor giroscópico para conseguir que el robot gire con precisión a un ángulo objetivo.

Pseudocode/Hints

Application	Objective	Error	Correction
Perro seguidor	Llegar a una posición objetivo desde la pared	Pulgadas o cm a la posición objetivo (posición_actual – posición_objetivo)	Velocidad en función de la distancia al objetivo
Seguidor de línea	Permanecer en el borde de la línea	Distancia de nuestra lectura de luz al borde de la línea (luz_actual – luz_objetivo)	Giro más agudo en función de la distancia a la línea
Sensor giroscópico	Giro a un ángulo objetivo	Número de grados hasta el ángulo objetivo	Giro más rápido en función de los grados restantes

Solución: Perro seguidor (sensor ultrasónico)



Solución: Seguidor de línea proporcional

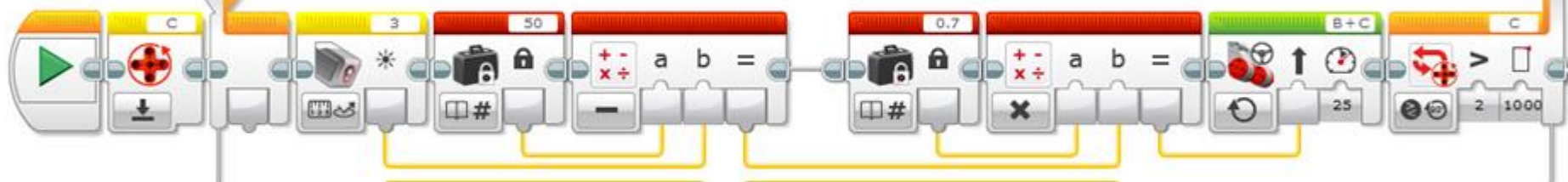
Nota: Este programa utiliza los sensores de color en el modo Luz reflejada. Esto significa que deberá calibrar los sensores. Por favor, lea nuestras lecciones de calibración antes de continuar! :-)

Recomendamos que su equipo utilice un seguidor de línea como éste. Es el más suave de los 4 seguidores de línea en esta lección. Hay incluso mejores seguidores de línea (que usan el control PID completo), pero uno que utiliza el control "P" ya es un buen comienzo. El seguidor de línea proporcional cambia el ángulo de giro en función de la distancia del robot a la línea.

Código escrito por Droids Robotics

01

Cada programa proporcional debe tener dos partes: Parte 1 que calcula el error (en este caso, distancia a la línea) y Parte 2 que calcula una corrección que es proporcional al error (en este caso cuánto debe girar). Podéis usar el control proporcional con otros sentidos también. Funciona muy bien!



Nota: no es necesario utilizar un bloque constante (50) con un cable de datos. En nuestra lección lo hemos hecho para dejar claro que hemos multiplicado por una constante.

Parte 1: Calcula el error
 - Nuestro objetivo es estar en el borde de la línea (sensor de luz = 50). El Bloque matemático de encima (-) calcula lo lejos que el robot está de nuestro objetivo de 50.
 - El bloque constante de arriba es nuestro objetivo. Podéis cambiarlo para diferentes tipos de líneas.
 - Tened en cuenta que en el peor de los casos, el sensor de luz leerá 0 ó 100 (Camino de la línea !!). Esto os dará un error = 50 ó -50.

Parte 2: Calcula y aplica el error
 - Multiplicamos el error de la Parte 1 por 0.7 para determinar el valor de giro.
 - Elegimos 0.7 de modo que cuando tengamos el peor de los casos (50 ó -50), la dirección del movimiento en el bloque de arriba será 35 ó -35, que es una curva cerrada.
 - Podéis ajustar este valor para que el seguidor de línea se ajuste a vuestras necesidades.

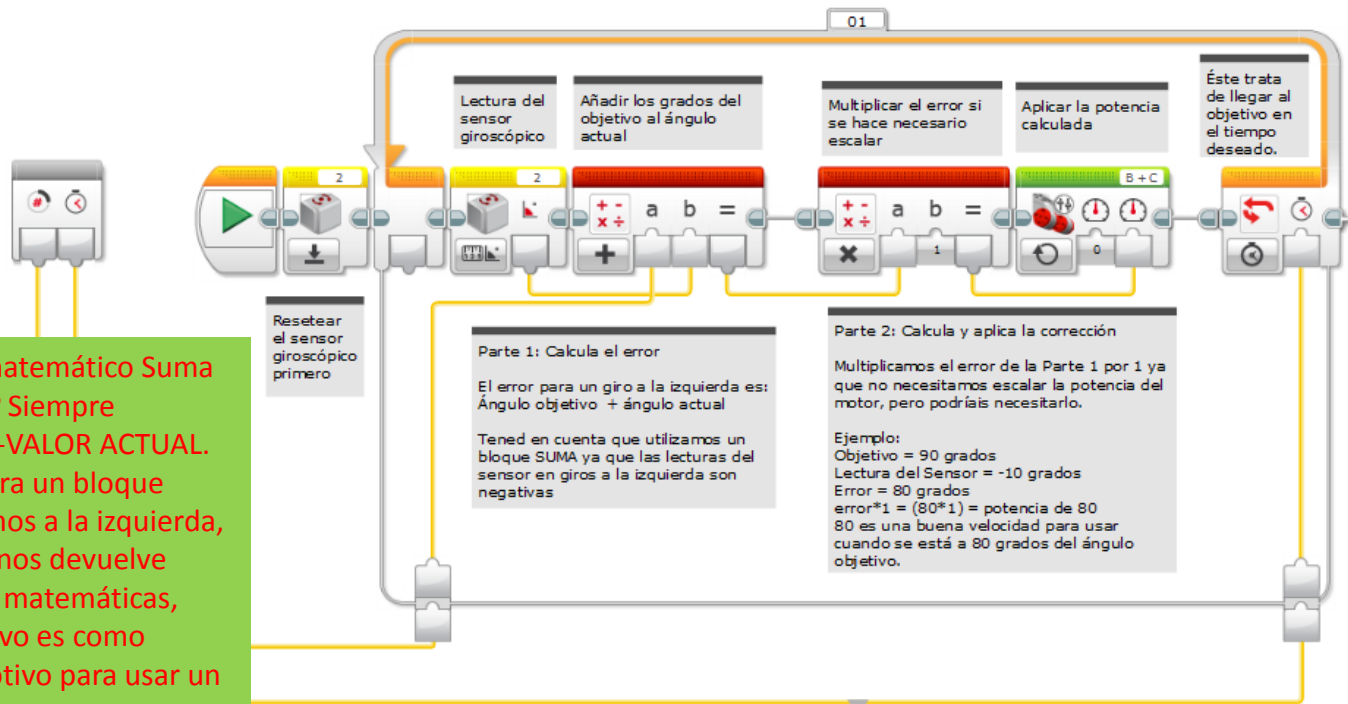
El seguidor de línea finaliza después de una rotación de 1000 grados. Ajustadlo a vuestras necesidades.

Solución: Giro a izquierda (sensor giroscópico)

El objetivo de este programa es la creación de un giro proporcional a izquierda que terminará después de un tiempo determinado. Gracias a Construction Mavericks por el código original que hemos modificado para esta lección! :-)

Bucle principal.
 ** IMPORTANTE - Giros a la izquierda hacen que el sensor giroscópico lea valores NEGATIVOS

- 1) leer el valor giroscópico
- 2) agragar el valor giroscópico al objetivo (ver nota anterior). Utilizar la escala si es necesario
- 3) introducir el resultado en la velocidad del motor derecho, manteniendo el motor izquierdo a 0.
- 4) repetir durante el tiempo especificado



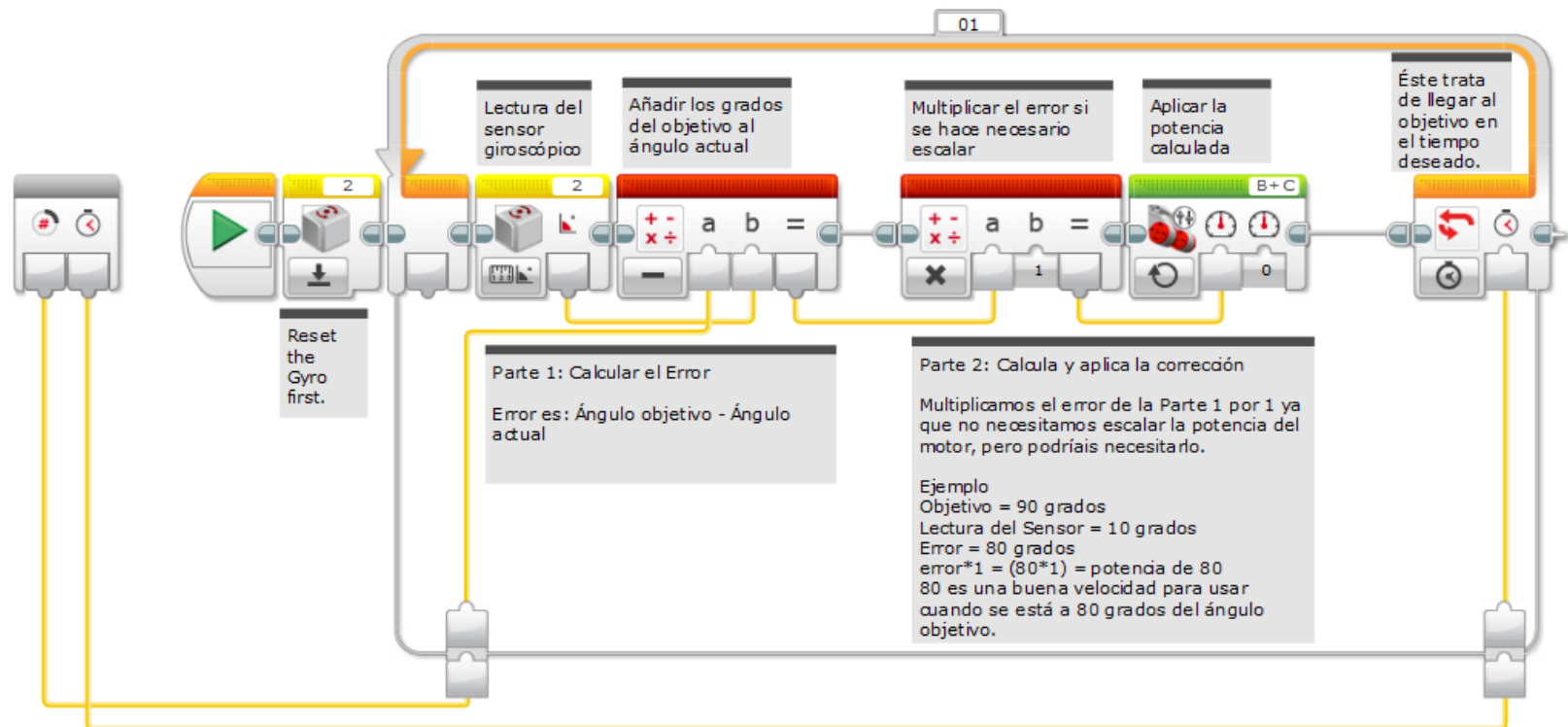
¿Por qué un bloque matemático Suma en el giro a izquierda? Siempre calculamos OBJETIVO-VALOR ACTUAL. Así que ¿por qué ahora un bloque SUMA? Cuando giramos a la izquierda, el sensor giroscópico nos devuelve valores negativos. En matemáticas, sumar un valor negativo es como restarlo. Éste es el motivo para usar un bloque SUMA.

Solución: Giro a derecha (sensor giroscópico)

El objetivo de este programa es la creación de un giro proporcional a derecha que terminará después de un tiempo determinado. Gracias a Construction Mavericks por el código original que hemos modificado para esta lección! :-)

This is the main turn loop.

- 1) leer el valor giroscópico
- 2) agregar el valor giroscópico al objetivo. Utilizar la escala si es necesario
- 3) introducir el resultado en la velocidad del motor derecho, manteniendo el motor izquierdo a 0.
- 4) repetir durante el tiempo especificado



Guía de discusión

1. ¿Qué significa Control Proporcional?

Respuesta. Movimiento más o menos rápido del robot en función de la distancia de éste al objetivo

2. ¿Qué tienen todos los códigos de Control Proporcional en común?

Respuesta. Todos calculan un error y aplican una corrección

Créditos

- Este tutorial ha sido creado por Sanjay Seshan y Arvind Seshan de Droids Robotics (team@droidsrobotics.org).
- El código original del sensor giroscópico ha sido proporcionado por Construction Mavericks (frank.levine@gmail.com)
- Traducción realizada por Toni Soler de Apps&Lego
- Más lecciones en www.ev3lessons.com



Este trabajo está bajo licencia [Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-nc-sa/4.0/).