PRÁCTICA II

Resolución de los ejercicios

Empezamos contando lo que tienen en común. En ambos casos todo el control del robot ocurre en la llamada `robobo.whenANewColorBlobIsDetected(blobDetectedCallback)`, y por tanto en la correspondiente función `blobDetectedCallback`, que es igual para los dos. Ahí, se centra el robot lo que haga falta para posteriormente avanzar. Ambas acciones se desglosan en correspondientes funciones `centerToAColor(blob)` y `moveToAColor(blob)`, la primera de ellas es compartida pero en la segunda ya difieren.

Ejercicio 1

Aquí `moveToAColor(blob)` implementa el control proporcional. Por tanto tenemos un coeficiente `KPav` y una variable `P` que es igual al sensor IR frontal. Con esto calculamos la corrección ε de forma que

$$v_t := v_{t-1} - \varepsilon_t$$

siendo v_t la velocidad del robot en el instante t.

Ejercicio 2

En dos cosas difiere del anterior. En primer lugar si bien anterior cuando `P` (la variable que carga el error actual) era menor que un cierto valor, la ejecución se paraba, ahora se llama a otra subrutina `blob_is_close(speed, distance)` que se encarga de girar el blob si esque estamos seguros de estar cerca, para luego para la ejecución.

La segunda cosa en la que difieren es la estategia de control, ahora cargamos con dos variables más (y sus correspondientes dos coeficientes), que se definen de la siguiente forma

$$I_t = \sum_{i \in [0,t]} P_i$$

$$D_t = P_t - P_{t-1}$$

con esto, definimos la nueva corrección:

$$\varepsilon_t := P_t \times K_p + I_t \times K_I + D_t \times K_D$$

Ajuste de controladores

Se han ajustato manualmente cambiado los valores en el programa y viendo el comportamiento en el simulador.

Pablo Chantada Saborido & José Romero Conde