

TRABAJO OBLIGATORIO 2

GR-08

Diego Domínguez Castillejo

Miguel Reino Mateos

Andrés Ripoll Cabrera

Índice:

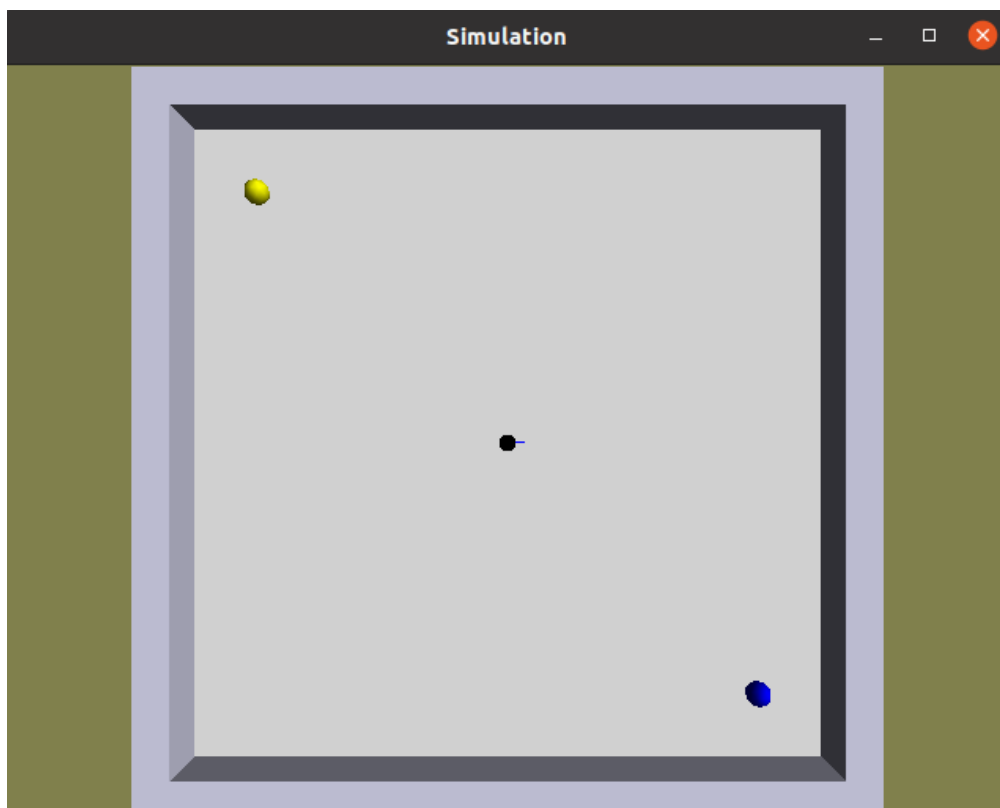
1. Introducción.....	3
2. Red neuronal estática	4
2.1. Esquema.....	4
2.2. Experimentos	5
3. Red neuronal dinámica	16
3.1. Esquema.....	16
3.2. Experimentos	17
4. Problemática y futuras implementaciones	24
4.1. Problemática	24
4.2. Futuras implementaciones	24
5. Conclusión.....	25
6. Anexo. Ficheros de parámetros.....	26

1. Introducción

La idea principal de nuestro trabajo pretende desarrollar un robot que, dependiendo del estado de su batería se acerque a una luz amarilla (batería baja) o a una azul (batería cargada).

Para desarrollar esta labor hemos implementado un ciclo de histéresis, y además hemos añadido una luz roja que se enciende cuando las luces anteriores se apagan, y cuya función es conseguir que el robot de vueltas a su alrededor.

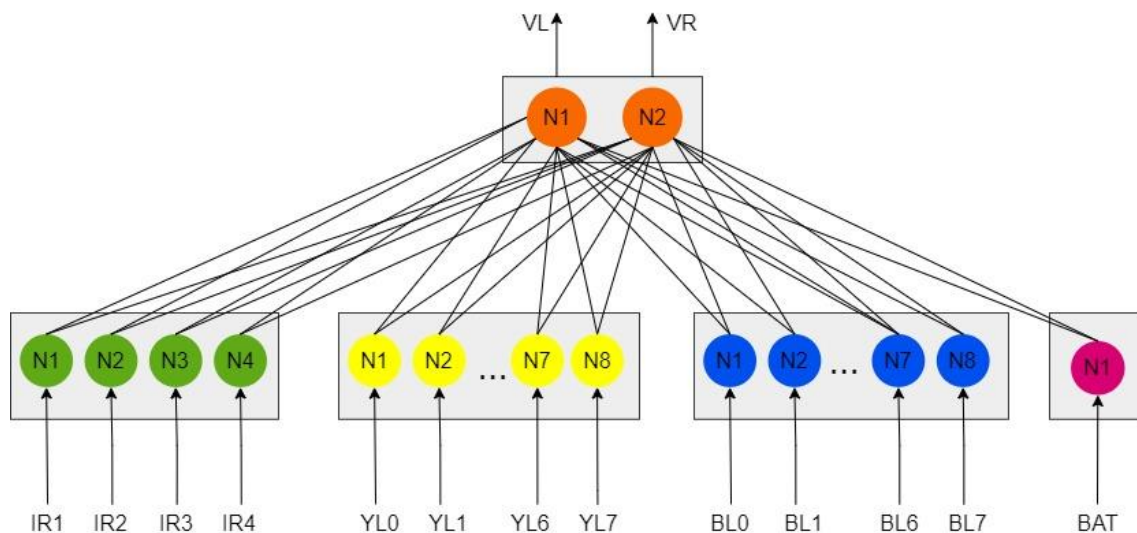
A lo largo de este documento detallaremos el tipo de red neuronal implementada, y los diferentes experimentos que hemos necesitado hasta alcanzar el mejor funcionamiento del robot para realizar la tarea descrita.



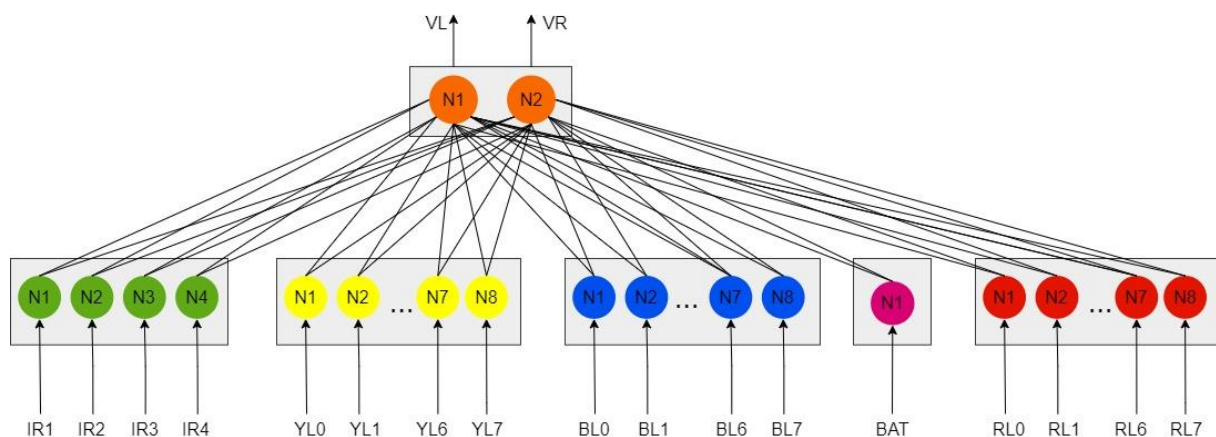
2. Red neuronal estática

2.1 Esquema

Inicialmente nuestro experimento solo contaba con sensores de luz amarilla (YLn) y azul (BLn), además del control de motores (VL y VR) y sensores de batería (BAT) y proximidad (IRn). En total, esta red contaba con 44 cromosomas ($4 \times 2 + 8 \times 2 + 8 \times 2 + 1 \times 2 + 2$). Esta arquitectura se empleará en los experimentos: punto de partida, 0, 1, 2 y 3.



Sin embargo, al aumentar la complejidad de la funcionalidad del robot, haciéndole girar alrededor de la luz roja, también tuvimos que añadir la nueva capa de los sensores de luz de este color (8×2), aumentando el total a 60. Esta arquitectura se empleará en los experimentos 4 y 5.



2.2 Experimentos

En esta sección vamos a desarrollar cada uno de los experimentos llevados a cabo, incluyendo la expresión de la función de fitness, su gráfica evolutiva y una breve explicación de porqué descartamos una determinada función para buscar otra más válida.

- **Punto de partida:**

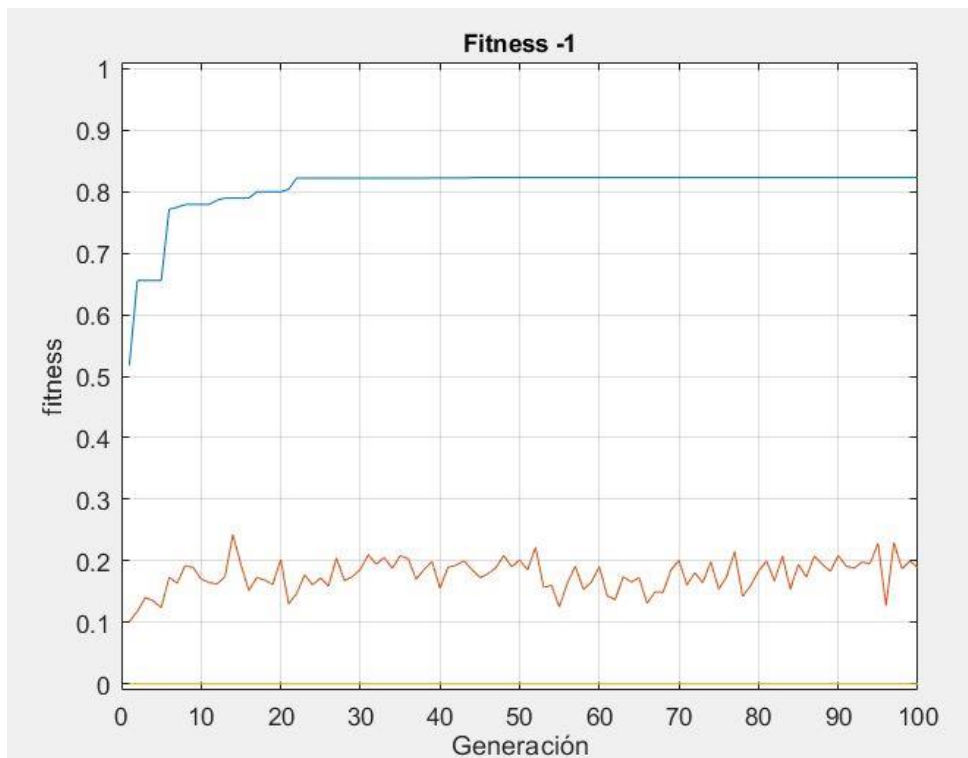
$$F = \frac{\sum_{i=0}^{N_{steps}} \left[\frac{1}{4} V(1 - \sqrt{\Delta v}) + \frac{3}{4} [BC_{on}(L_0 + L_7) \max\{L_i\} + BC_{off}(BL_0 + BL_7) \max\{BL_i\}] \right]}{N_{steps}} \left(1 - \min\left\{ \frac{N_{coll}, 10}{10} \right\} \right)$$

La función de fitness de la que partimos es esta, que consta principalmente de 3 partes: la primera corresponde a la máxima velocidad recta, como se ha visto en clase; la segunda, que corresponde a la trayectoria hacia el punto de carga (BC_{on} , luz amarilla) desde el punto de descarga (BC_{off} , luz azul), y la tercera parte se refiere al número de colisiones que permite la función, visto también en clase.

Inicialmente definimos un umbral de batería a 0,3. Este umbral irá variando a medida que vaya a una luz u otra.

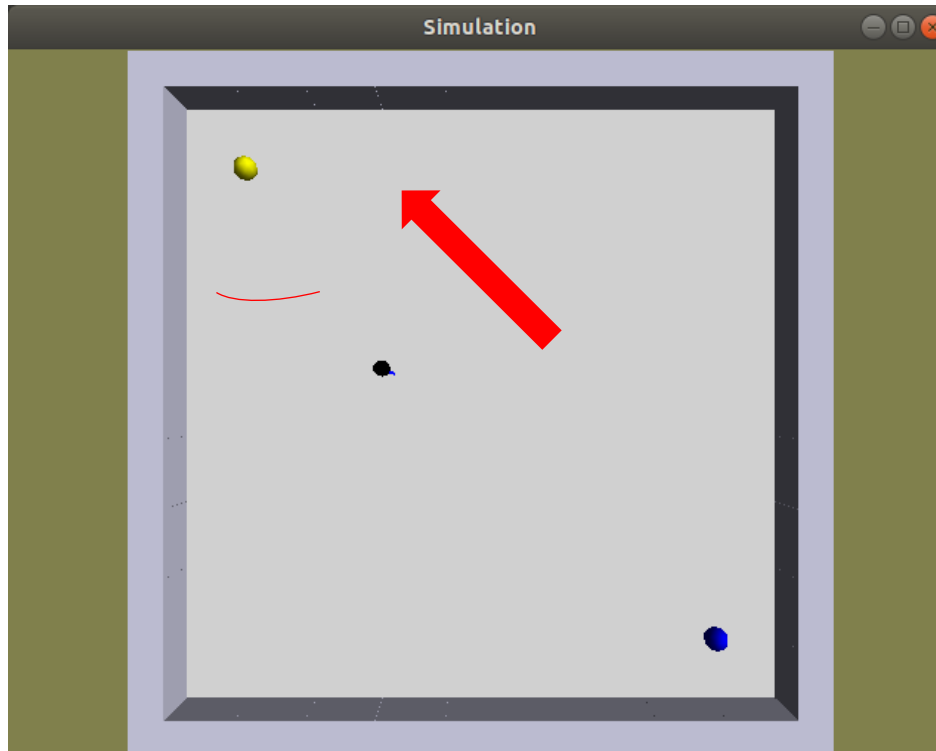
$$BC \begin{cases} BC_{on} & \text{if } Bat \leq Bat_{umbral} ; \quad Bat_{umbral} = 0,8 \\ \\ BC_{off} & \text{if } Bat > Bat_{umbral} ; \quad Bat_{umbral} = 0,3 \end{cases}$$

A continuación, mostramos la gráfica de la función de fitness:



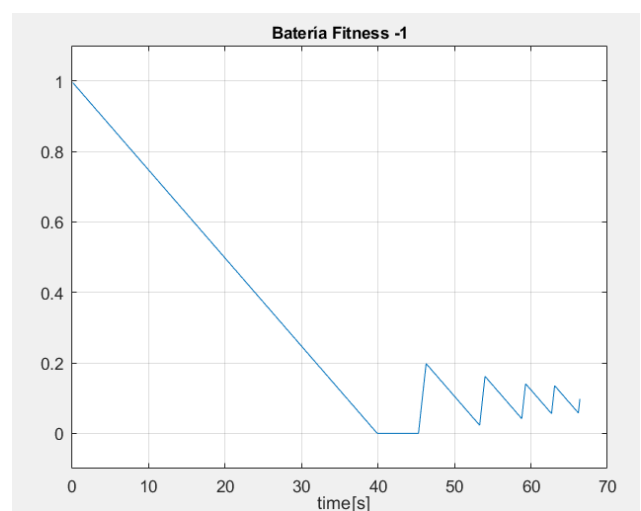
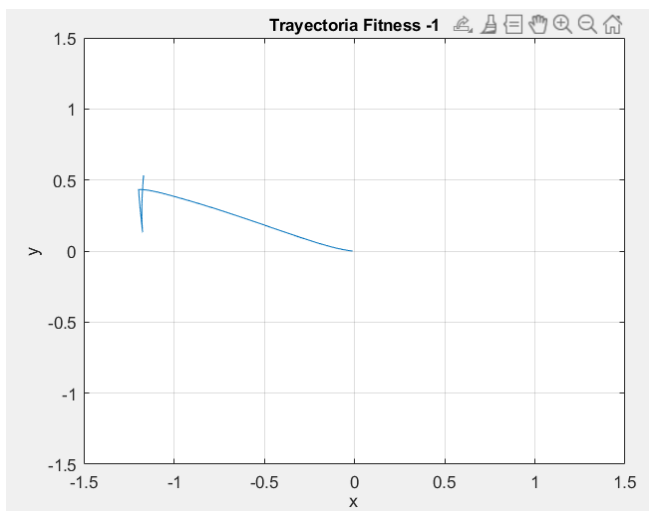
A pesar de su elevado valor, su funcionalidad se aleja claramente de lo que queremos. Esto es debido a que no se penaliza la función fitness con el sensor batería, provocando errores en el funcionamiento del robot.

También incluimos una imagen que resume el movimiento del robot:



Aunque en esta imagen pueda parecer que el robot funciona correctamente, en realidad estaba yendo a la luz amarilla de espaldas. Además se quedaba estancado en la región entre la zona de carga y descarga (arco rojo). A pesar del valor elevado de la fitness, claramente este experimento no cumple con su finalidad ideada.

De manera adicional, mostramos las gráficas de nivel de batería y posición del robot:



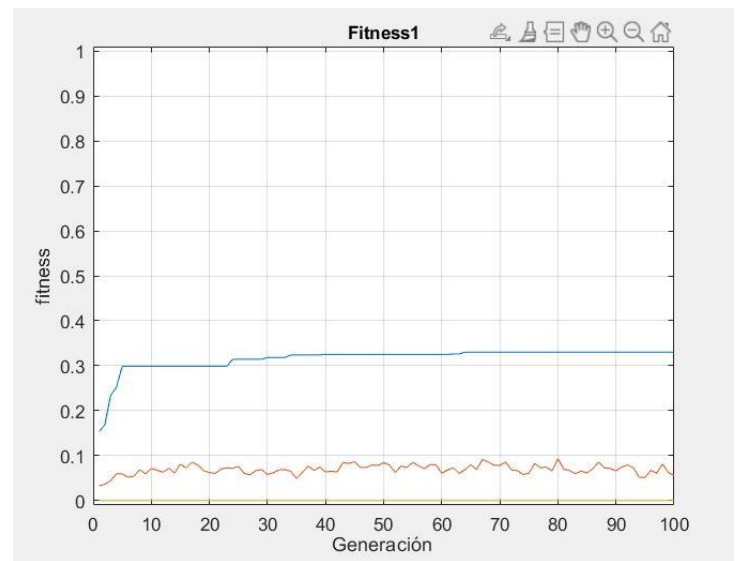
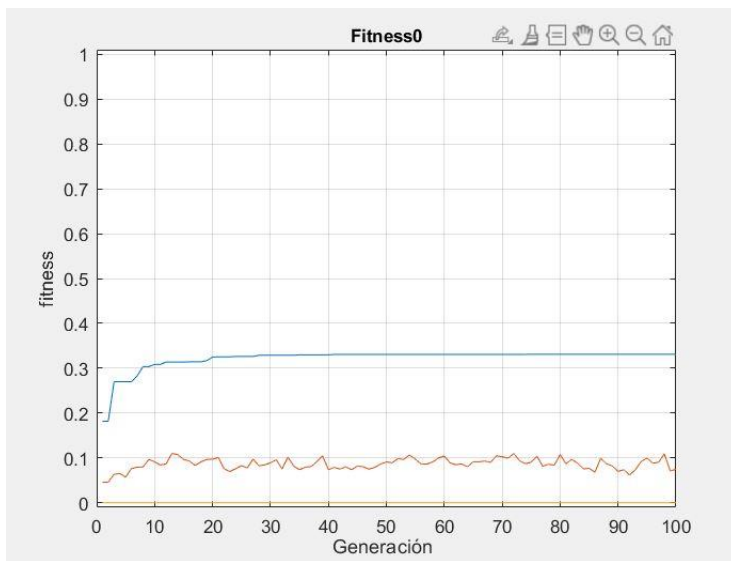
Como se puede observar, el movimiento del robot no es el esperado y la batería no se carga correctamente.

- **Fitness 0 y 1:**

$$F = \frac{\sum_{i=0}^{N_{steps}} \left[\frac{1}{4} V(1 - \sqrt{\Delta v}) + \frac{3}{4} [BC_{on}(L_0 + L_7) \max\{L_i\} Bat + BC_{off}(BL_0 + BL_7) \max\{BL_i\}(1 - Bat)] \right]}{N_{steps}} \left(1 - \min\left\{ \frac{N_{coll}, 10}{10} \right\} \right)$$

En este experimento hemos añadido un factor que penaliza a la función, concretamente el sensor batería, que multiplica a los sensores de luz.

La función es la misma para ambos experimentos, la diferencia entre ellos es que en el experimento 0 empleamos una mutación del 15% y del 20% para el 1. Este cambio consigue que la trayectoria del robot sea más directa y definida.

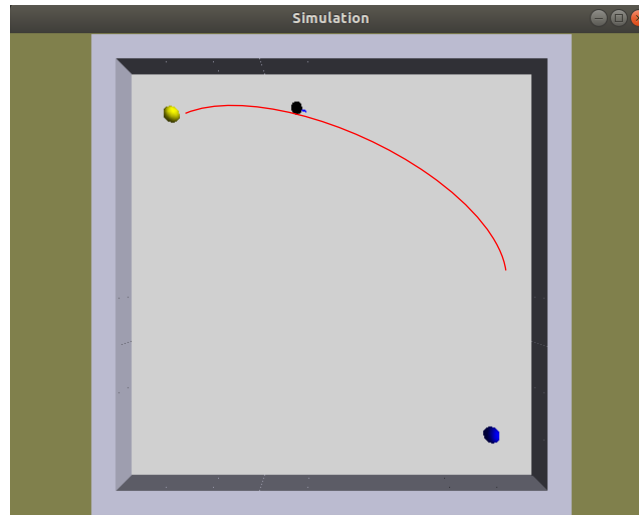


El reducido valor de estas dos fitness se debe principalmente a que la expresión de fitness utilizada en estos experimentos hace uso de la batería como penalización, como factor multiplicativo. Sin embargo, si fuera un valor aditivo el valor de fitness aumentaría, como haremos en el siguiente experimento.

Comparando estas gráficas con la obtenida en el apartado anterior, llegamos a la conclusión de que la fitness empeora, aunque el movimiento del robot si que es el esperado, capaz de oscilar entre las dos luces.

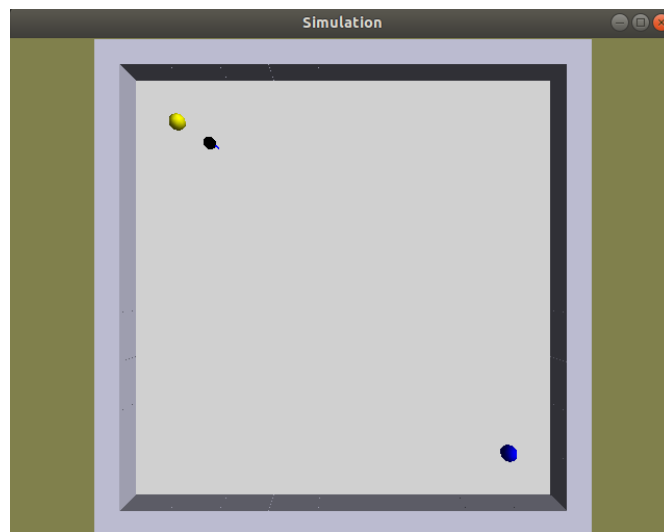
Vamos a describir este movimiento con las siguientes capturas.

Experimento 0



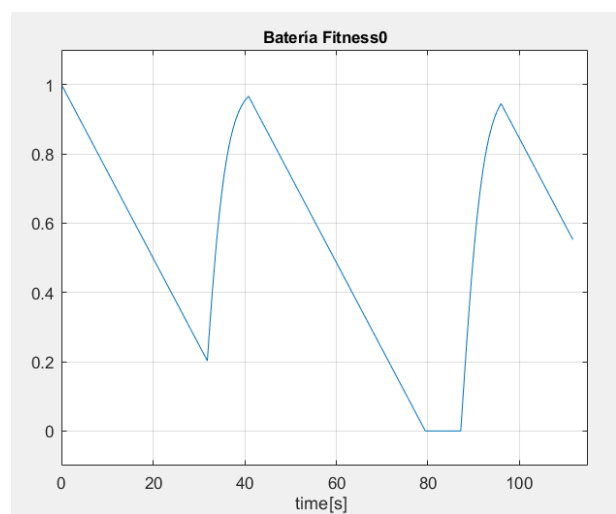
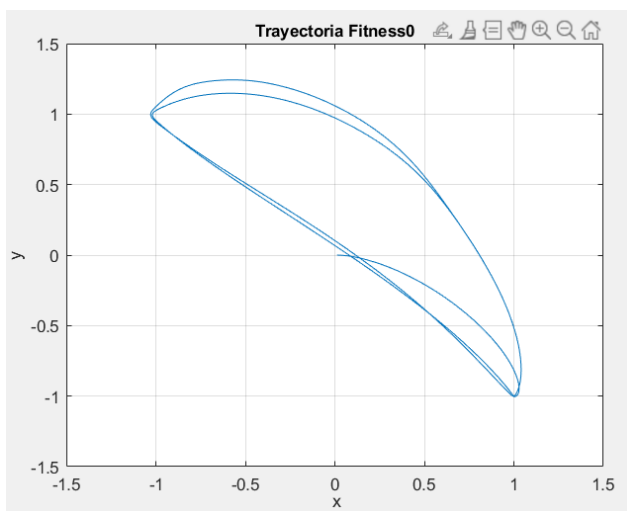
El experimento 0 sigue una trayectoria similar a la dibujada, demasiado exagerada, por lo que decidimos cambiar la mutación.

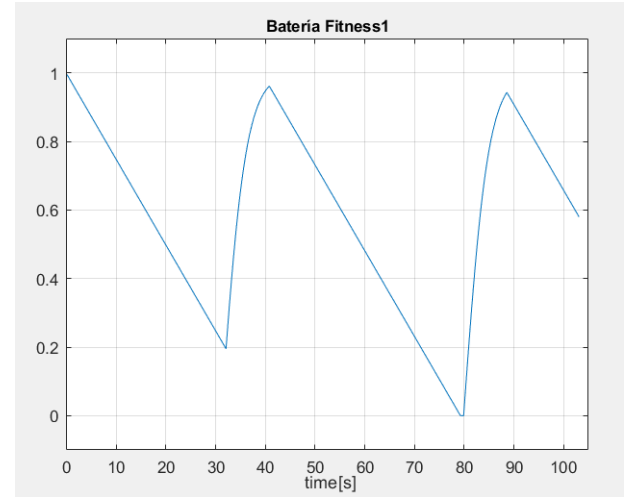
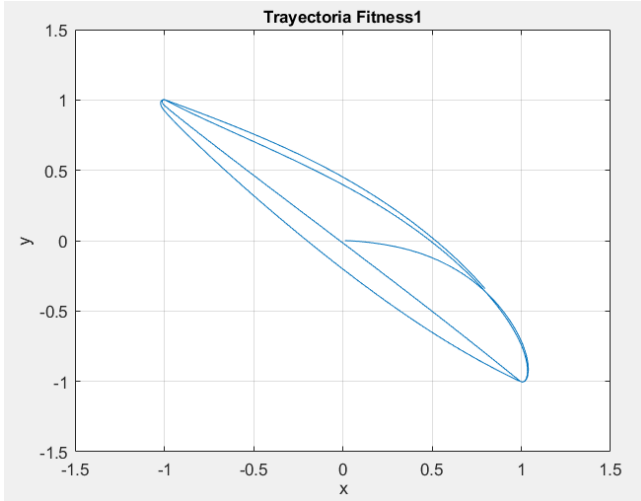
Experimento 1



En cambio, en el experimento 1 ahora la trayectoria es mucho más directa.

Para visualizar mejor este movimiento adjuntamos las gráficas de la trayectoria y del sensor de batería de ambos experimentos.

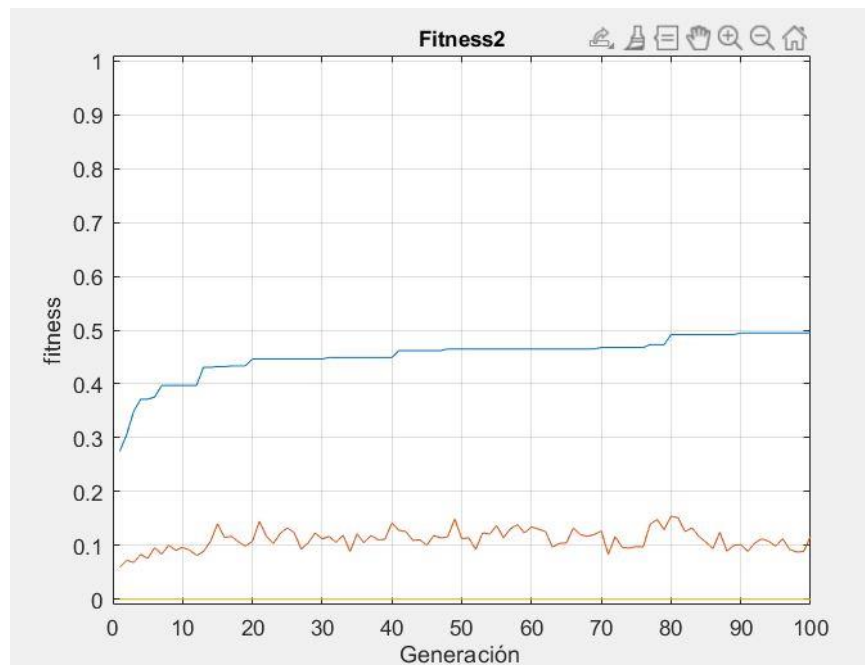




- **Fitness 2:**

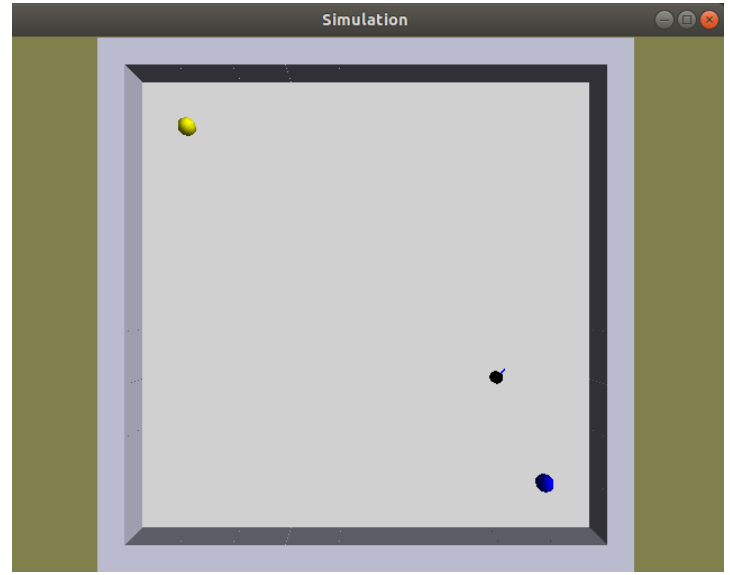
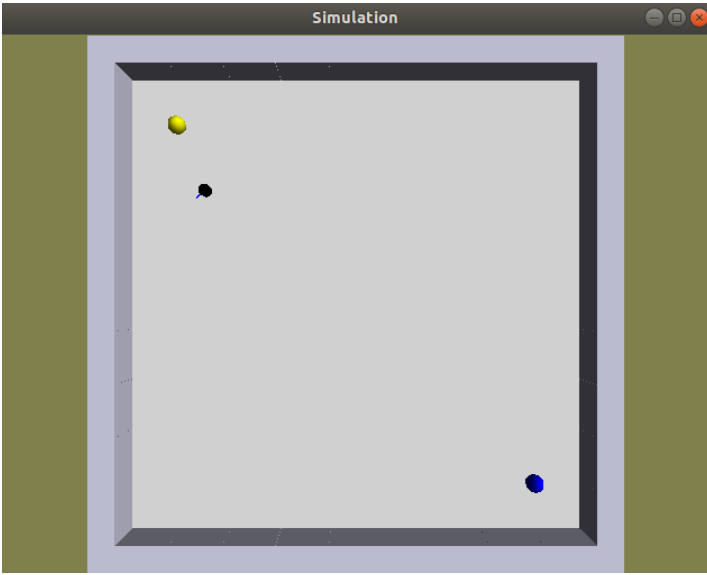
$$F = \frac{\sum_{i=0}^{N_{steps}} \left[\frac{1}{4} V(1 - \sqrt{\Delta v}) + \frac{3}{4} \left[BC_{on} \frac{\left(\frac{L_0 + L_7}{2} \right) + Bat}{2} + BC_{off} \frac{\left(\frac{BL_0 + BL_7}{2} \right) + (1 - Bat)}{2} \right] \right]}{N_{steps}} \left(1 - \min \left\{ \frac{N_{coll}}{10}, 10 \right\} \right)$$

Este experimento se implementa mediante la suma aritmética de los componentes, pero sin incluir la intensidad luminosa. Como se puede apreciar, se promedian los sensores de luz, se hace la media con el sensor batería, y se suman ambos comportamientos:



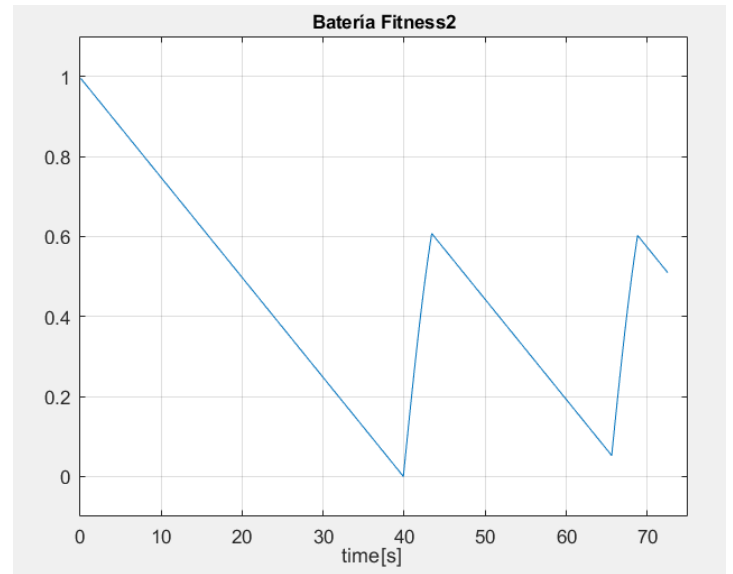
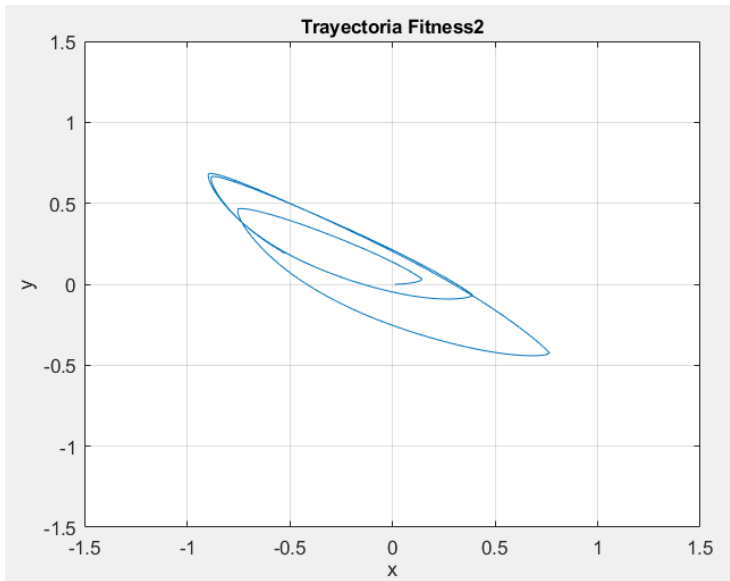
Como ya hemos comentado, ahora el factor de batería se introduce mediante una suma, por lo que la fitness no es tan reducida como en los casos anteriores.

En la siguiente imagen vemos las dos posiciones entre las que oscila nuestro robot:



Este movimiento puede ser mejorado, y es lo que haremos en el siguiente experimento.

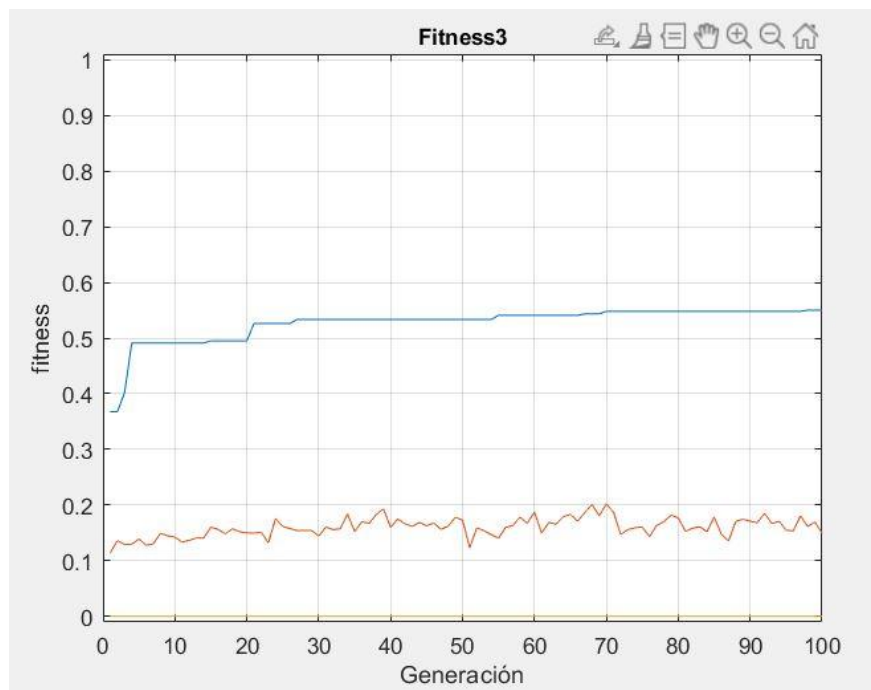
Observando las siguientes gráficas se puede observar como la trayectoria del robot es similar a los anteriores experimentos pero al carecer de la máxima intensidad luminosa, azul o amarilla, este recorrido es más corto. Por cómo está definido su comportamiento, la batería se carga y descarga más rápidamente.



- **Fitness 3:**

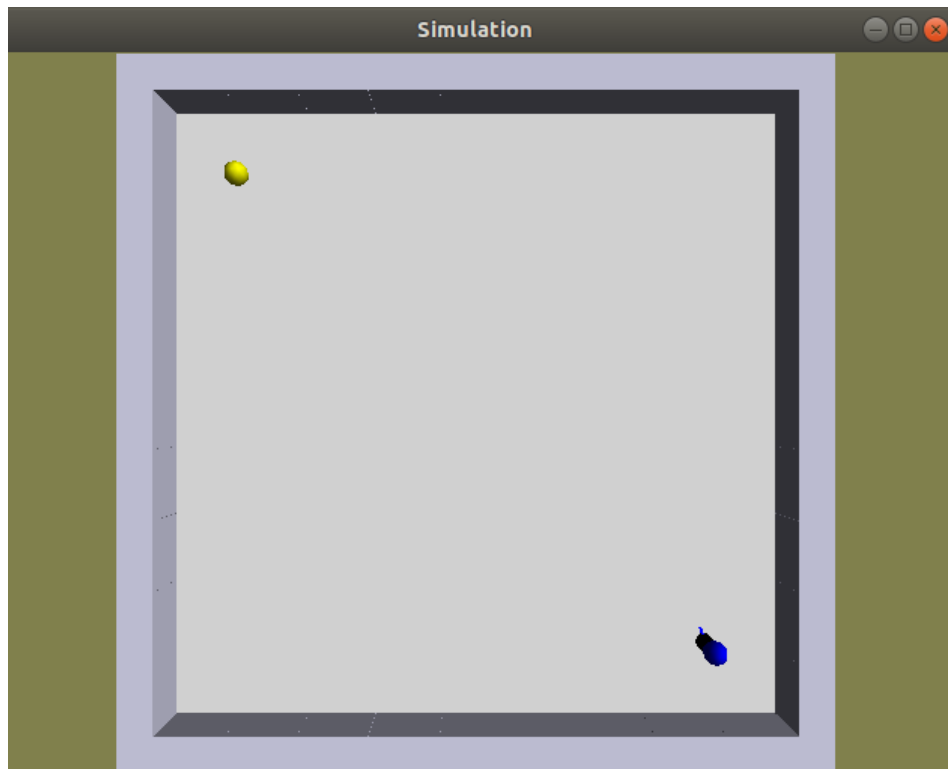
$$F = \frac{\sum_{i=0}^{N_{steps}} \left[\frac{1}{4} V(1 - \sqrt{\Delta v}) + \frac{3}{4} \left[BC_{on} \frac{\left(\frac{L_0 + L_7}{2} \right) + Bat + \max\{L_i\}}{3} + BC_{off} \frac{\left(\frac{BL_0 + BL_7}{2} \right) + (1 - Bat) + \max\{BL_i\}}{3} \right] \right]}{N_{steps}} \left(1 - \min \left\{ \frac{N_{coll}}{10} \right\} \right)$$

Esta fitness es parecida al anterior, pero sí incluye la intensidad luminosa. Respecto al experimento anterior, la trayectoria del robot cambia, acercándose más a las luces, y además, su fitness mejora bastante, llegando más rápidamente al valor máximo del experimento 2 y tomando un valor final cercano a 0,6.

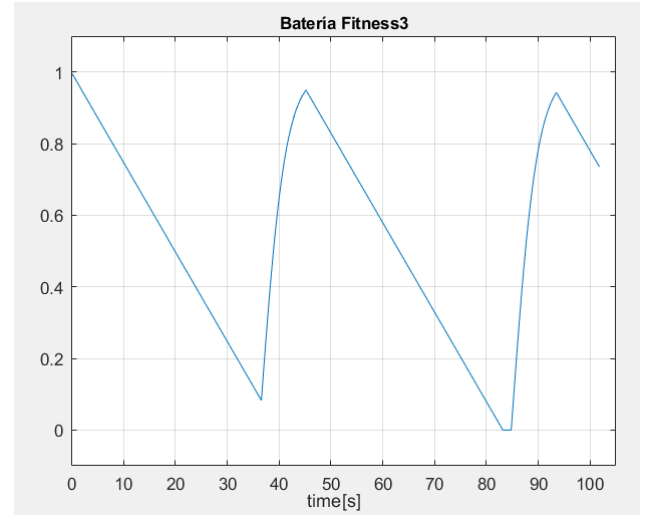
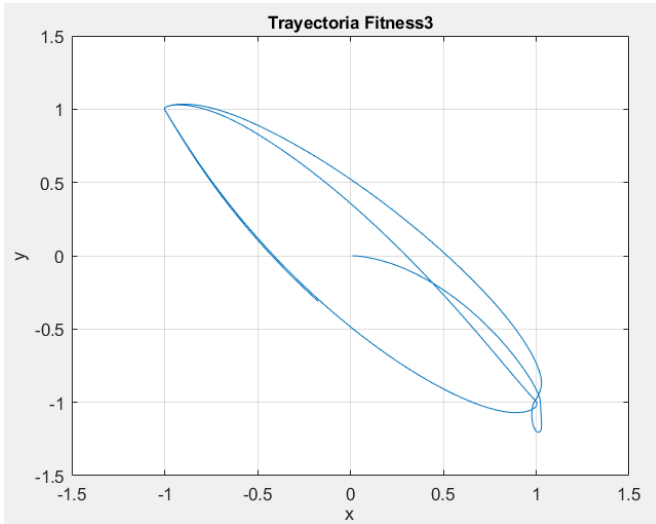


Se puede apreciar que el valor de esta fitness es ligeramente superior al anterior debido al valor de máxima intensidad de luz, ya que al acercarnos más a la luz obtendremos mejores resultados.

El recorrido de este experimento es similar al de los experimentos 0 y 1, pero describiendo una órbita más ovalada, sin seguir una línea recta.



En las siguientes gráficas podemos ver mejor la trayectoria del movimiento del robot y como se descarga y carga la batería.

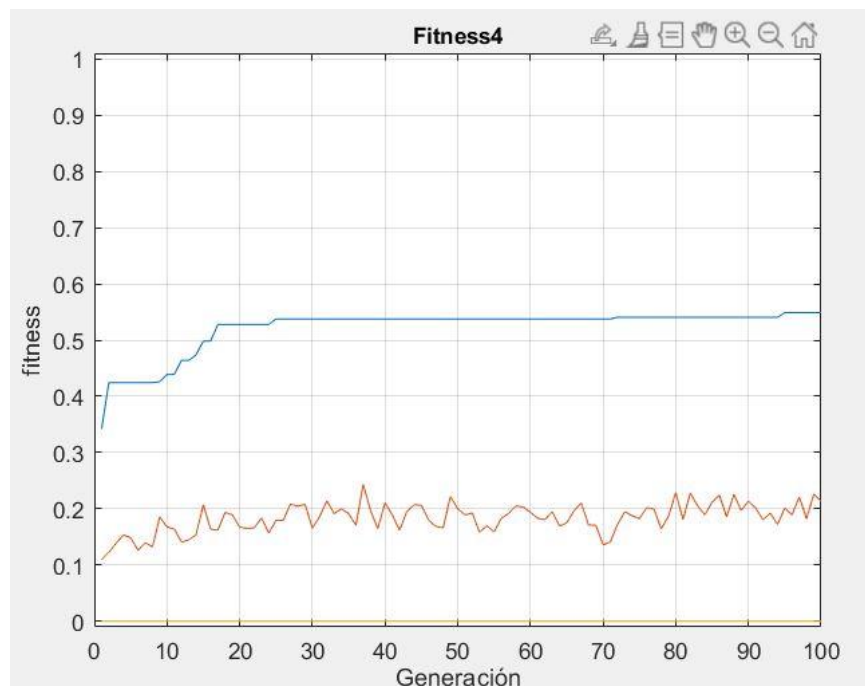


- **Fitness 4:**

$$F = \frac{\sum_{i=0}^{N_{steps}} \left[\frac{1}{4} V(1 - \sqrt{\Delta v}) + \frac{3}{4} \left[RL_{off} BC_{on} \frac{\left(\frac{L_0 + L_7}{2} \right) + Bat + \max\{L_i\}}{3} + RL_{off} BC_{off} \frac{\left(\frac{BL_0 + BL_7}{2} \right) + (1 - Bat) + \max\{BL_i\}}{3} + RL_{on} \frac{\left(\frac{RL_0 + RL_7}{2} \right) + \max\{RL_i\}}{2} \right] \right]}{N_{steps}} \left(1 - \min \left\{ \frac{N_{coll}, 10}{10} \right\} \right)$$

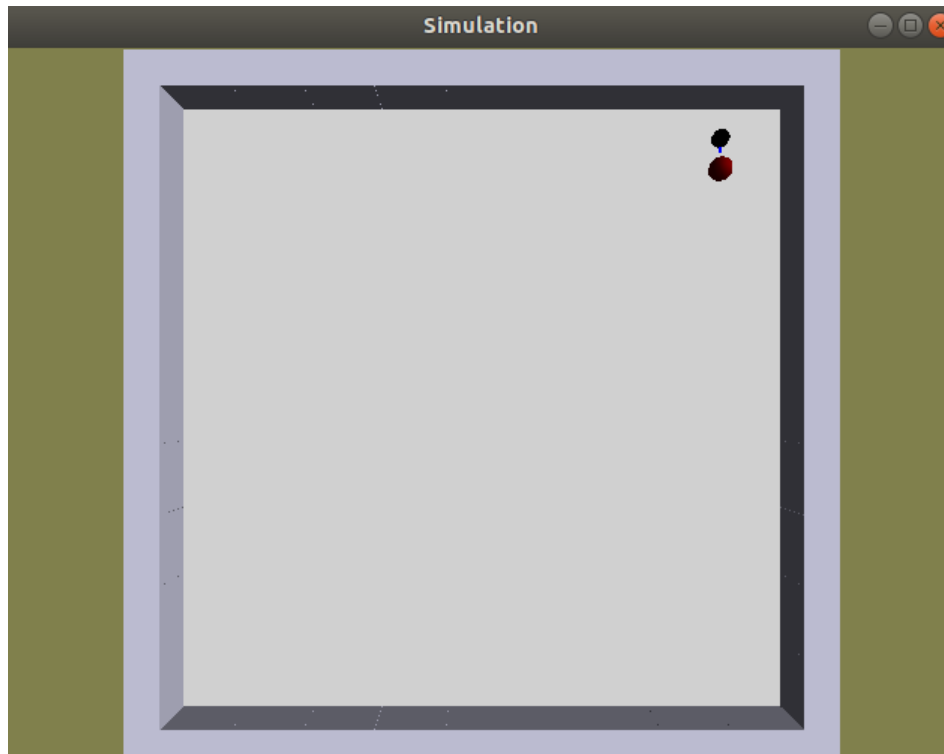
En el experimento 4 añadimos una luz roja que aparecerá cada 500 ms, apagando el resto de luces. La nueva fitness implementada hará desplazarse al robot hacia la luz roja. Hay que tener en cuenta que se utiliza un factor RL que depende del comportamiento que tenga el robot.

$$RL = \begin{cases} RL_{on} & \text{if } \exists \text{ Red light} \\ RL_{off} & \text{if } \nexists \text{ Red light} \end{cases}$$

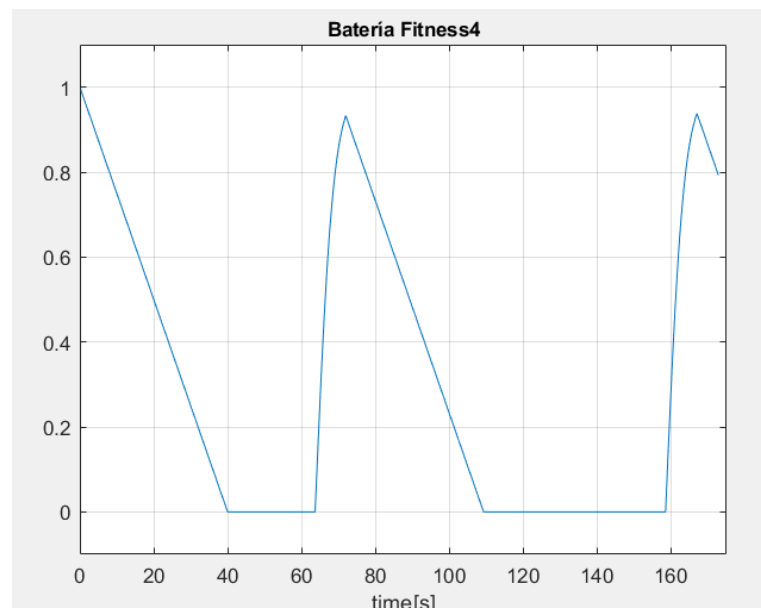
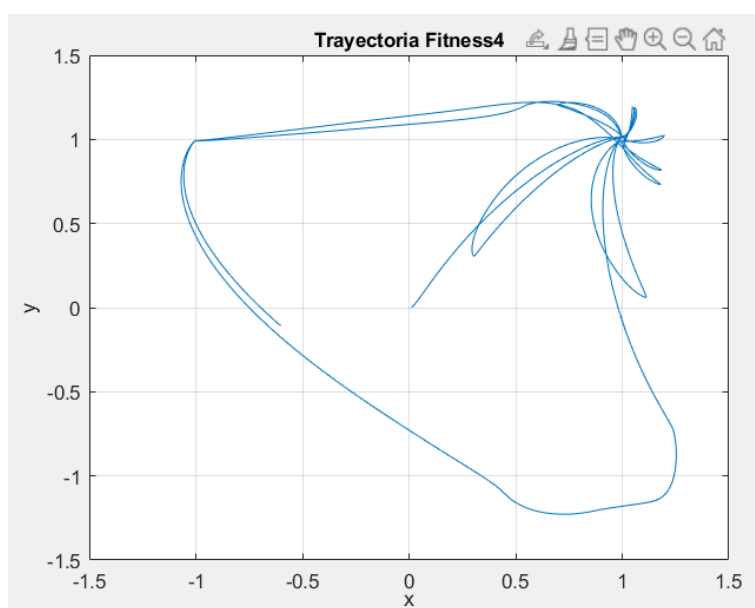


Aunque esta nueva función de fitness incluye el movimiento hacia la luz roja, su valor máximo no cambia en gran medida respecto al experimento anterior, esto se debe a que, al dejar un poco de lado la existencia de batería, hay una parte del recorrido que nos penaliza severamente.

Vemos como ahora el robot atraviesa la luz roja:



En la gráfica de la trayectoria, vemos como atraviesa la luz roja, en un recorrido parecido a una hélice o un molino. Sin embargo, la gráfica de batería no es del todo ilustrativa porque nuestro principal objetivo es que supiera adaptarse al estado de luz roja o luz azul y amarilla.

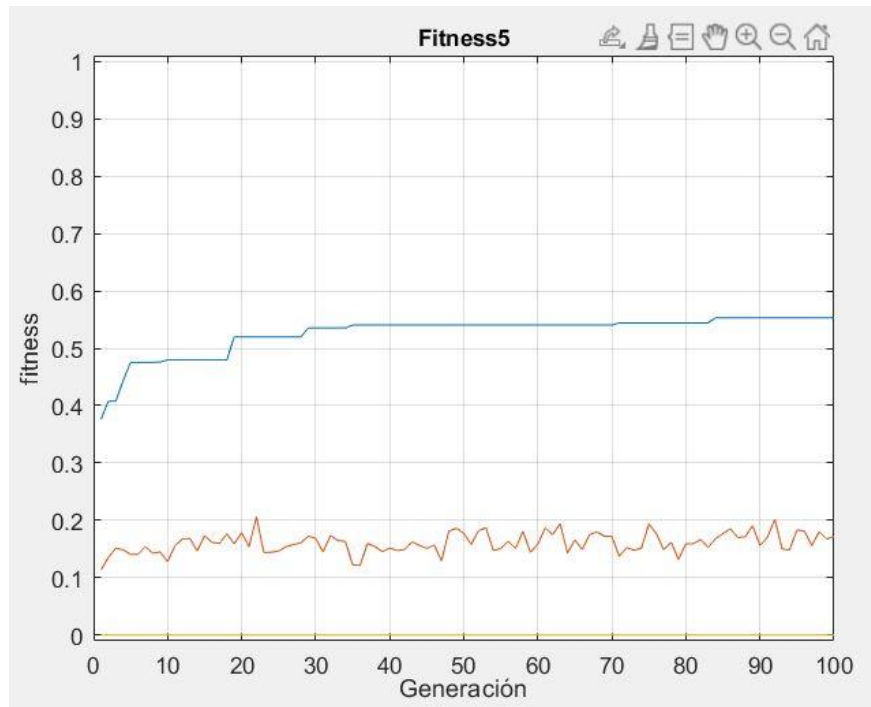


- **Fitness 5:**

$$F = \frac{\sum_{i=0}^{N_{steps}} \left[\frac{1}{4} V(1 - \sqrt{\Delta v}) + \frac{3}{4} \left[RL_{off} BC_{on} \frac{(L_0 + L_7)}{2} + Bat + \max\{L_i\} + RL_{off} BC_{off} \frac{(BL_0 + BL_7)}{2} + (1 - Bat) + \max\{BL_i\} + RL_{on} \frac{(RL_7 + RL_6)}{2} + \max\{RL_i\} \right] \right]}{N_{steps}} \left(1 - \min\left\{ \frac{N_{coll}, 10}{10} \right\} \right)$$

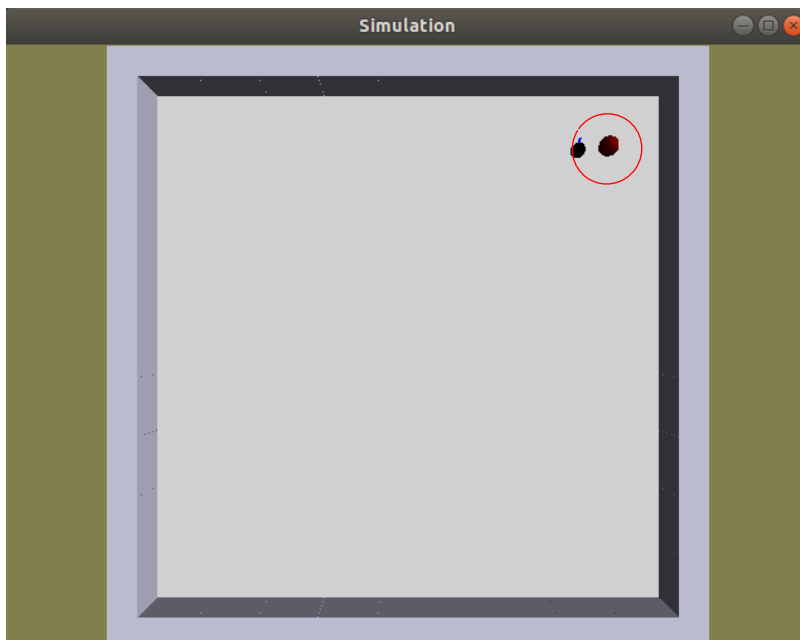
En este último experimento, cambiamos en la función fitness los sensores con los que se detecta la luz roja, siendo para este caso el sensor 6 y 7. De esta manera conseguimos que el robot dé vueltas alrededor de la luz roja.

La fitness definitiva quedaría:

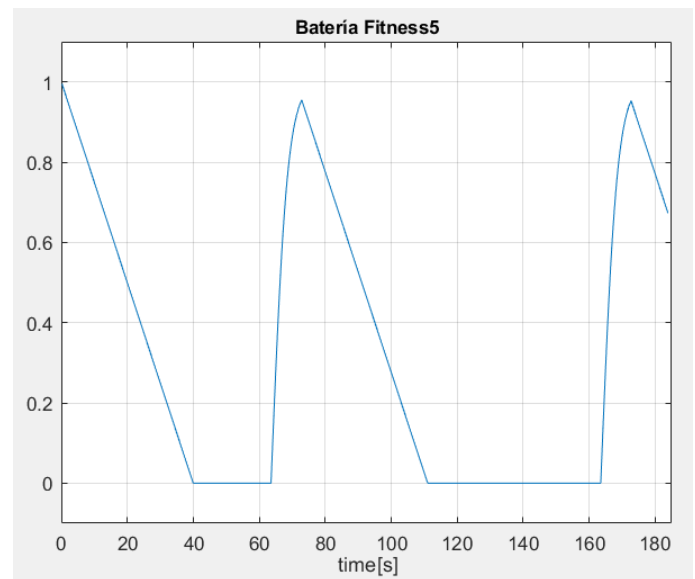
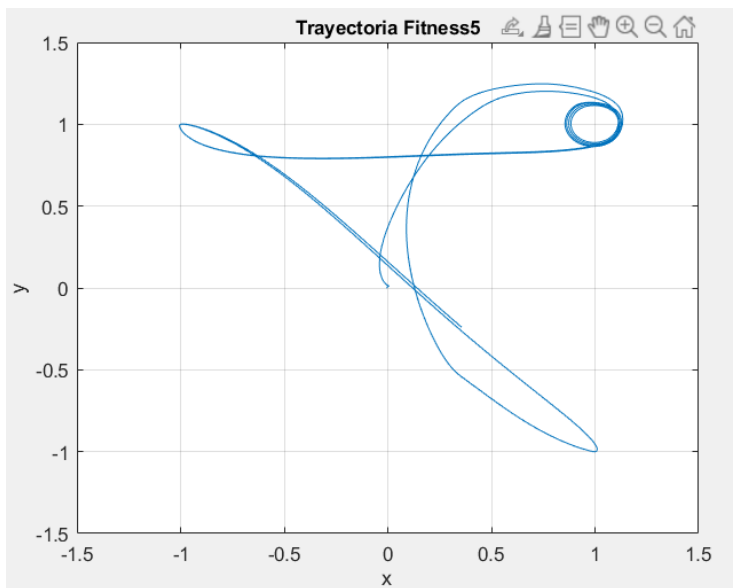


Esta última fitness guarda un cierto parecido con la anterior porque, como ya hemos comentado, el único cambio de comportamiento es la utilización de los sensores RL_6 y RL_7 .

En esta simulación del experimento se ve como el robot esta orientado para dar vueltas alrededor de la luz roja.



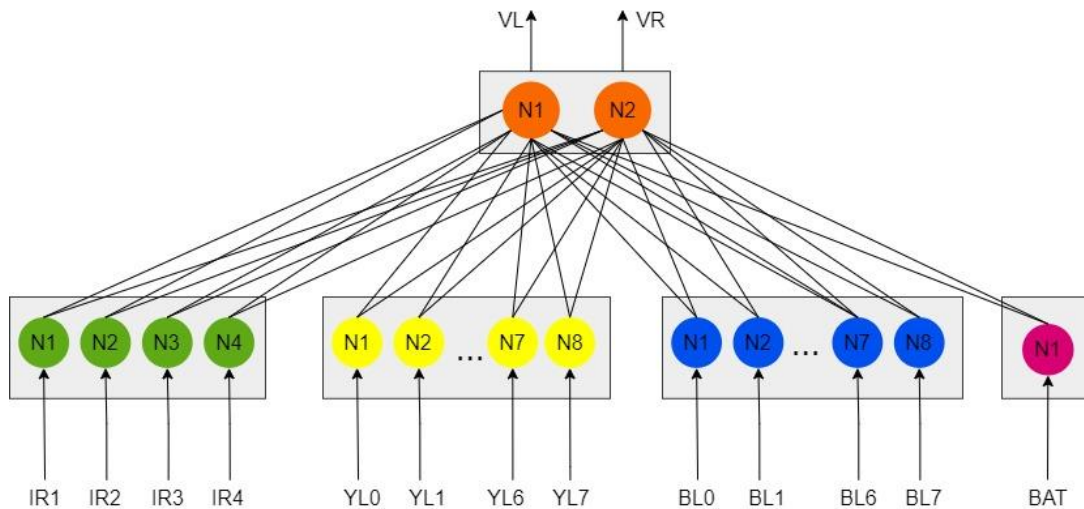
Estas son las gráficas definitivas de la trayectoria y el sensor batería para este experimento. Como se ha destacado en el anterior experimento, la gráfica del sensor batería no es del todo orientativa.



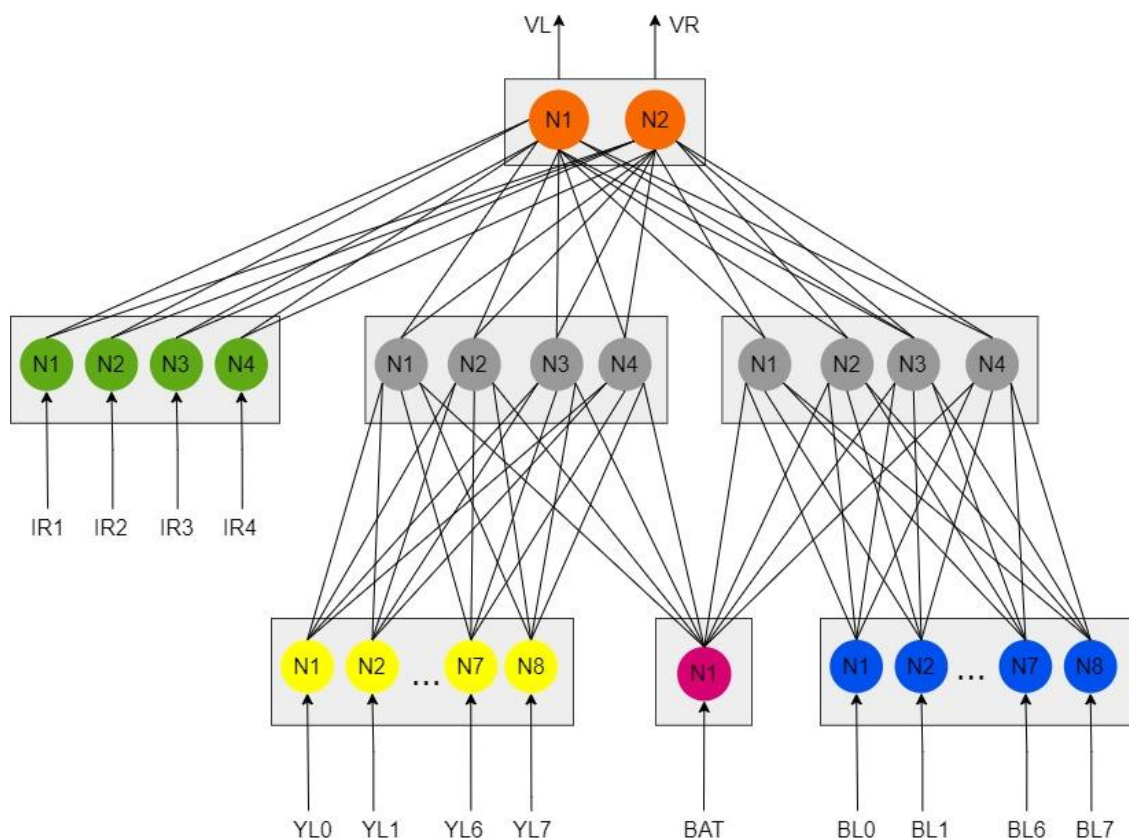
3. Red neuronal dinámica

3.1 Esquema

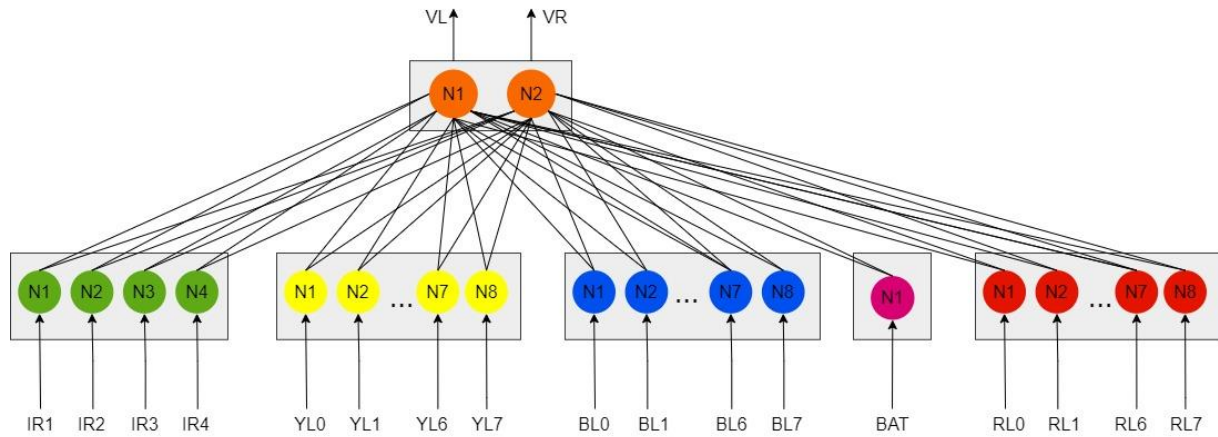
Inicialmente partiremos de la primera red neuronal empleada en la estructura estática, pero el número de cromosomas total será de 48. Esta se empleará en los experimentos 6 y 8:



Después, emplearemos una nueva red neural. Para esta nueva red vamos a añadir una nueva capa asociativa, que junta los sensores de luz amarilla y batería por un lado, y de luz azul y batería por otro lado, haciendo un total de 126. Esta se empleará únicamente en el experimento 7:



Finalmente, volveremos a emplear la segunda red neuronal de la estructura estática, cuyo número de cromosomas será aumentado a 64. Esta se empleará en el experimento número 9:

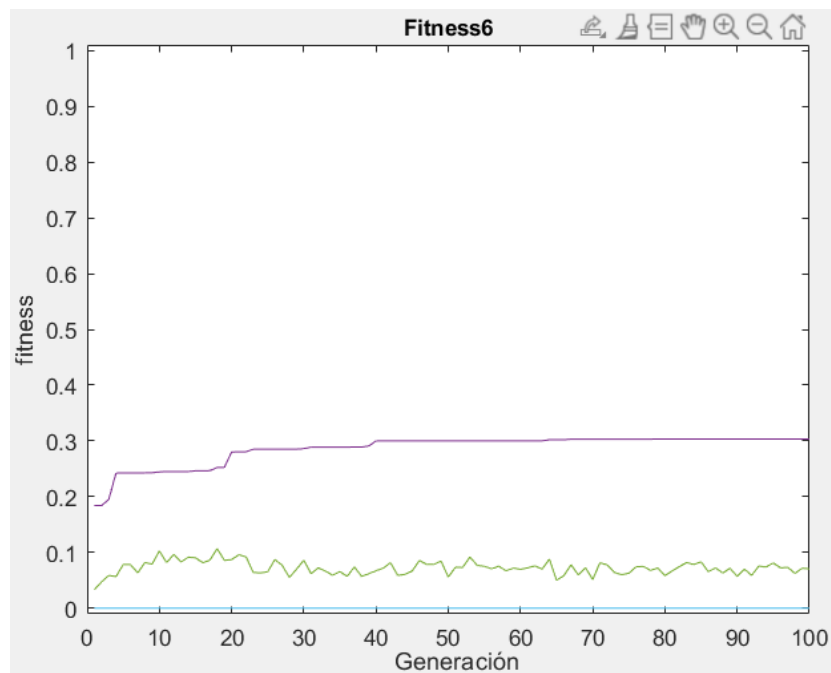


3.2 Experimentos

- Fitness 6:**

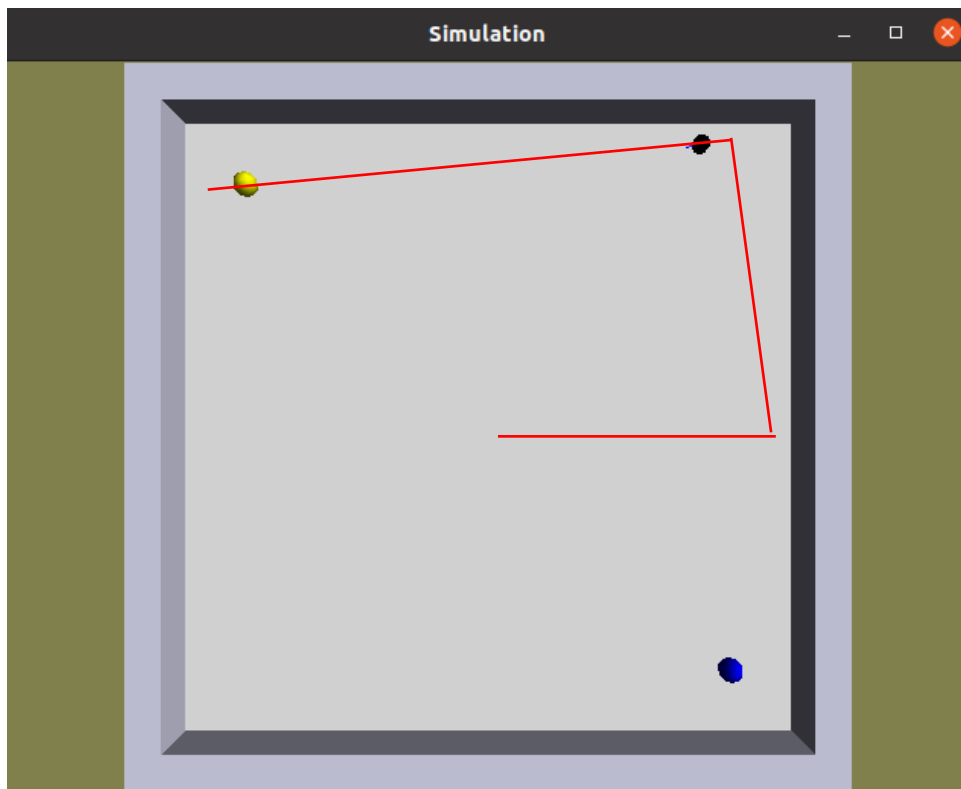
$$F = \frac{\sum_{i=0}^{N_{steps}} \left[\frac{1}{4} V(1 - \sqrt{\Delta v}) + \frac{3}{4} [BC_{on}(L_0 + L_7) \max\{L_i\} Bat + BC_{off}(BL_0 + BL_7) \max\{BL_i\}(1 - Bat)] \right]}{N_{steps}} \left(1 - \min\left\{ \frac{N_{coll}, 10}{10} \right\} \right)$$

En este experimento retomamos la función fitness del experimento número 0 y 1. Al incluir una arquitectura dinámica podemos ver como la fitness es un poco peor que en el caso estático, aunque apenas se aprecia cambio.

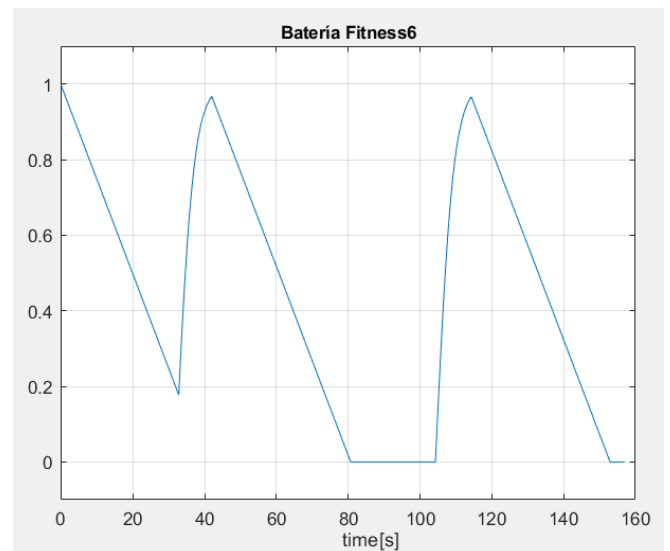
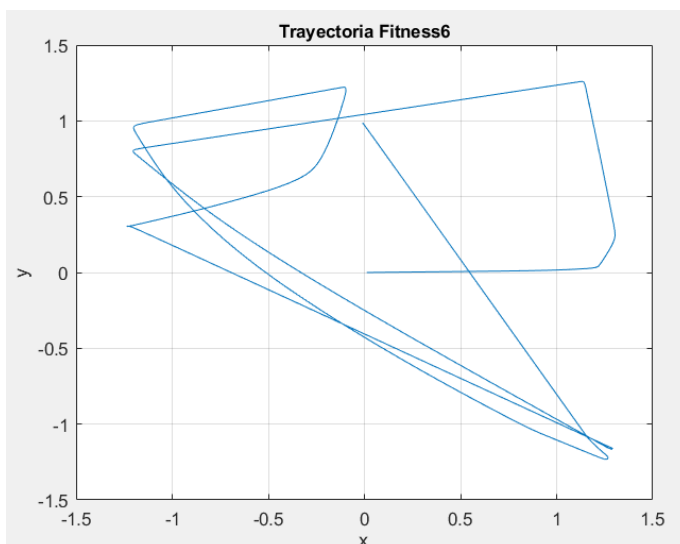


A esta función de fitness le ocurre algo parecido a lo visto en los primeros experimentos, la batería como factor penalizante multiplicativo reduce su valor considerablemente, situación que no pasa con medias aritméticas que veremos a continuación.

Sin embargo, en la simulación si que podemos apreciar un gran cambio en el comportamiento del robot. El robot no se dirige directamente hacia la luz, sino que recurre a rebotes en las paredes hasta encontrar una orientación apropiada para encontrar las luces.



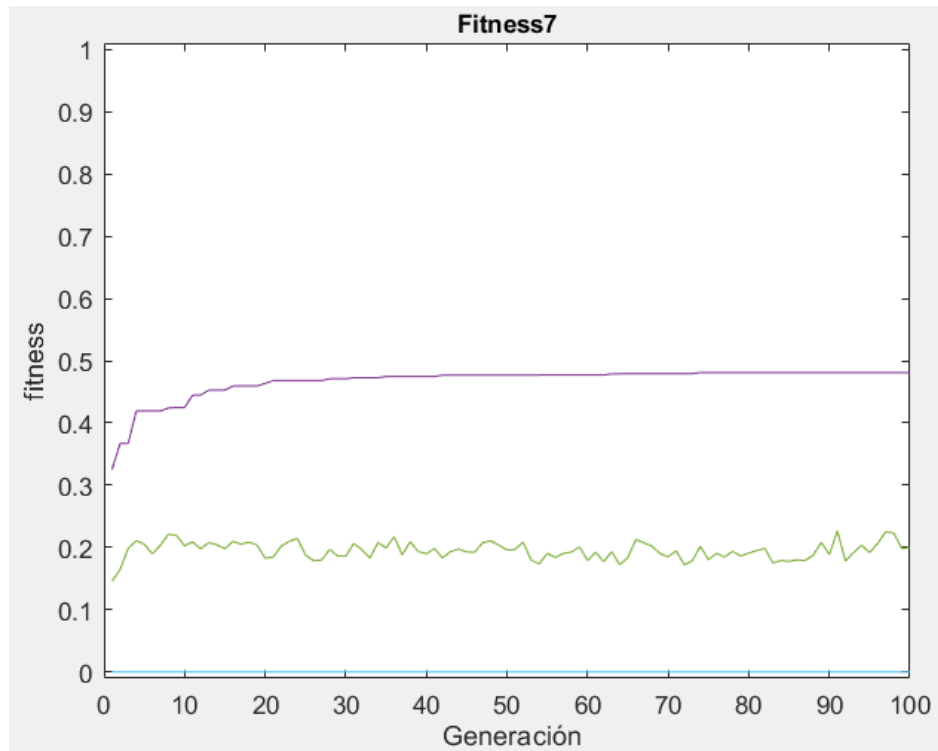
La gráfica de la trayectoria describe mejor este recorrido, donde el robot va rebotando contra las paredes de la simulación. La gráfica de la batería nos indica cuando el robot se recarga, con una pendiente más pronunciada que cuando se descarga.



- **Fitness 7:**

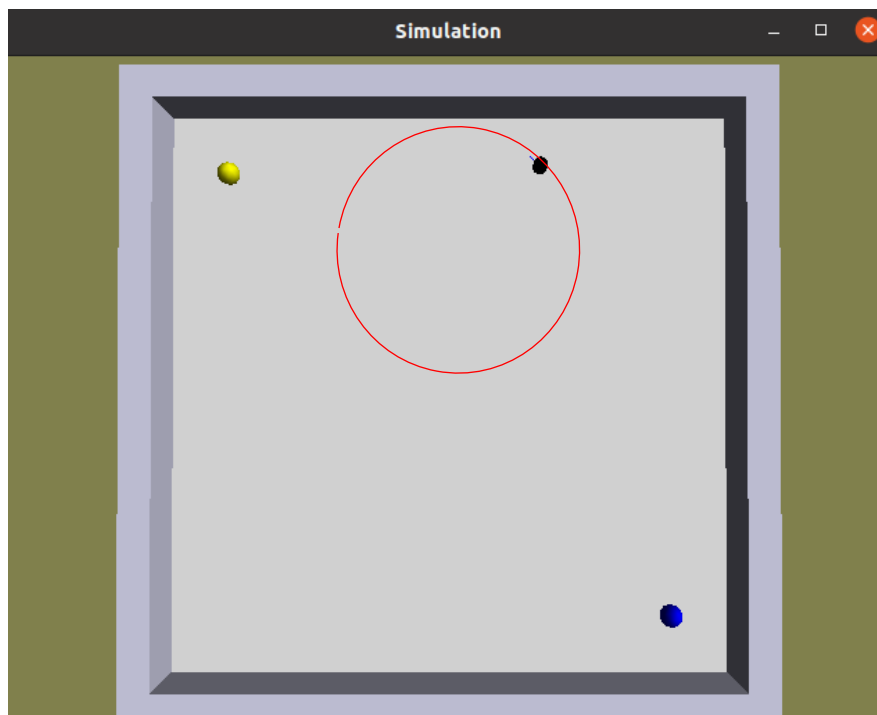
$$F = \frac{\sum_{i=0}^{N_{steps}} \left[\frac{1}{4} V(1 - \sqrt{\Delta v}) + \frac{3}{4} \left[BC_{on} \frac{\left(\frac{L_0 + L_7}{2} \right) + Bat + \max\{L_i\}}{3} + BC_{off} \frac{\left(\frac{BL_0 + BL_7}{2} \right) + (1 - Bat) + \max\{BL_i\}}{3} \right] \right]}{N_{steps}} \left(1 - \min\left\{ \frac{N_{coll}, 10}{10} \right\} \right)$$

La fitness de este experimento será la misma que la empleada en el experimento número 3. Volviendo a comparar las fitness sacamos las mismas conclusiones que para el apartado anterior.

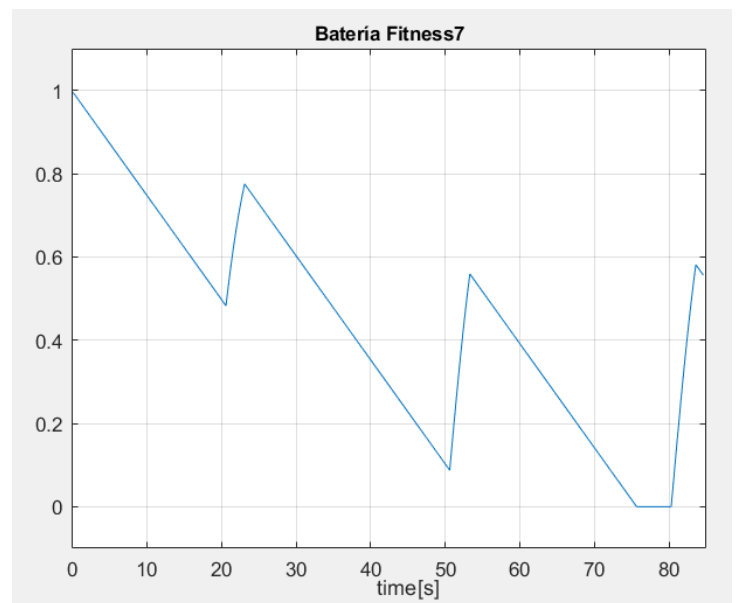
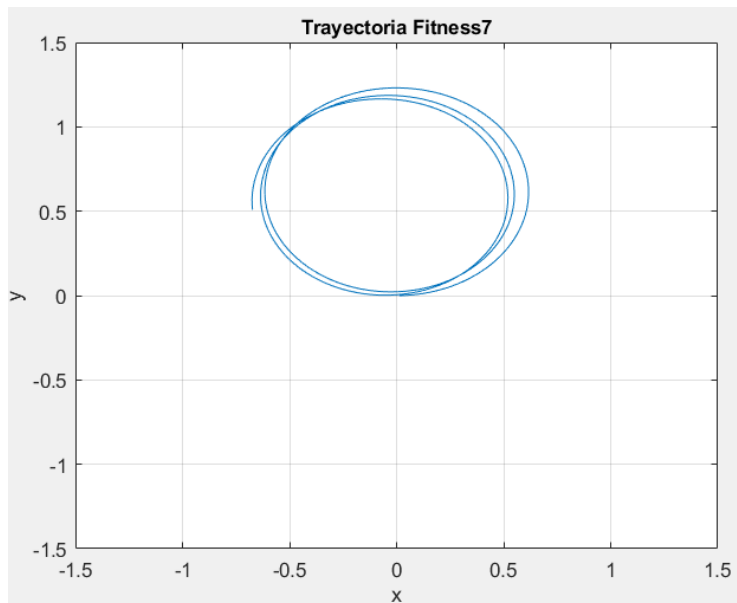


Como era de esperar, por cómo está definida la función de fitness actual mediante la suma de los sensores, el valor crece de nuevo.

Tanto en este experimento como en el que sigue a este el robot no realiza el recorrido esperado, y simplemente da vueltas describiendo la circunferencia que viene dibujada en la captura.



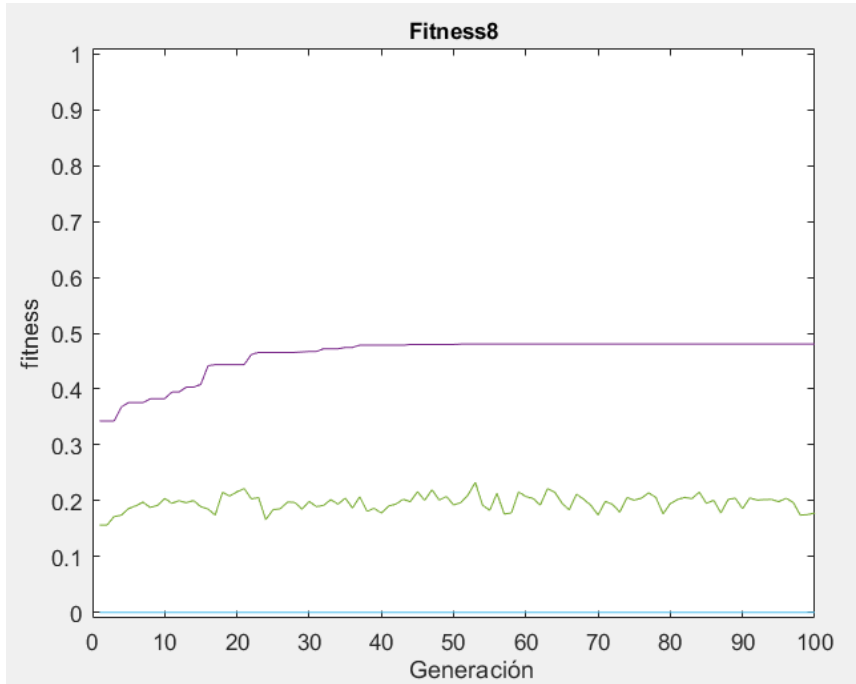
Con estas dos gráficas se puede observar mejor este recorrido, ya que la trayectoria es lo que describíamos previamente, y la batería se estabiliza para 54 s, ya que la trayectoria es siempre la misma.



- **Fitness 8:**

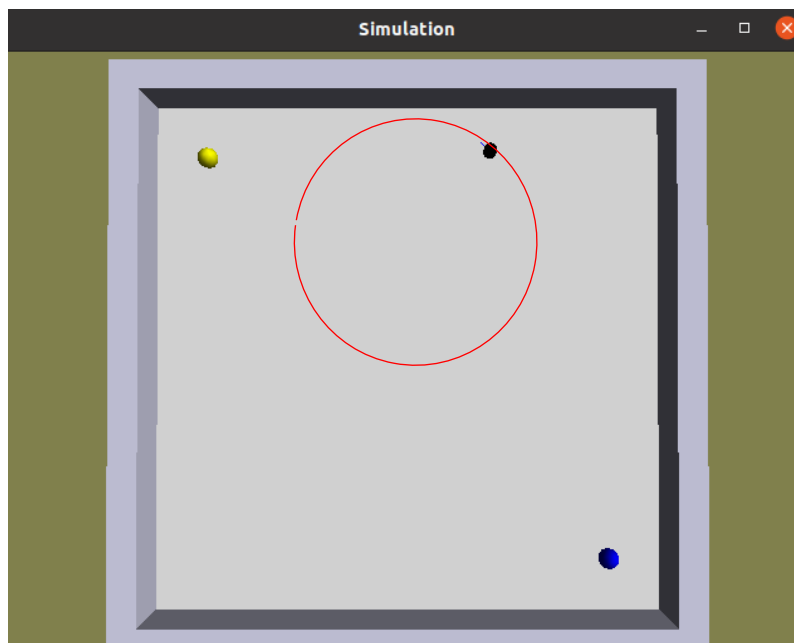
$$F = \frac{\sum_{i=0}^{N_{steps}} \left[\frac{1}{4} V(1 - \sqrt{\Delta v}) + \frac{3}{4} \left[BC_{on} \frac{\left(\frac{L_0 + L_7}{2} \right) + Bat + \max\{L_i\}}{3} + BC_{off} \frac{\left(\frac{BL_0 + BL_7}{2} \right) + (1 - Bat) + \max\{BL_i\}}{3} \right] \right]}{N_{steps}} \left(1 - \min\left\{ \frac{N_{coll}}{10} \right\} \right)$$

Para el desarrollo de este experimento simplemente tomamos el anterior y le implementamos la nueva arquitectura híbrida. El resultado en la función de fitness es similar al anterior.

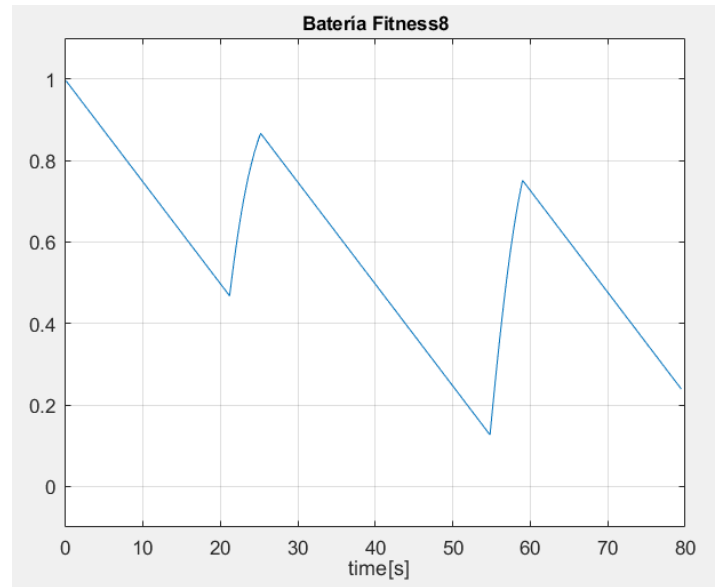
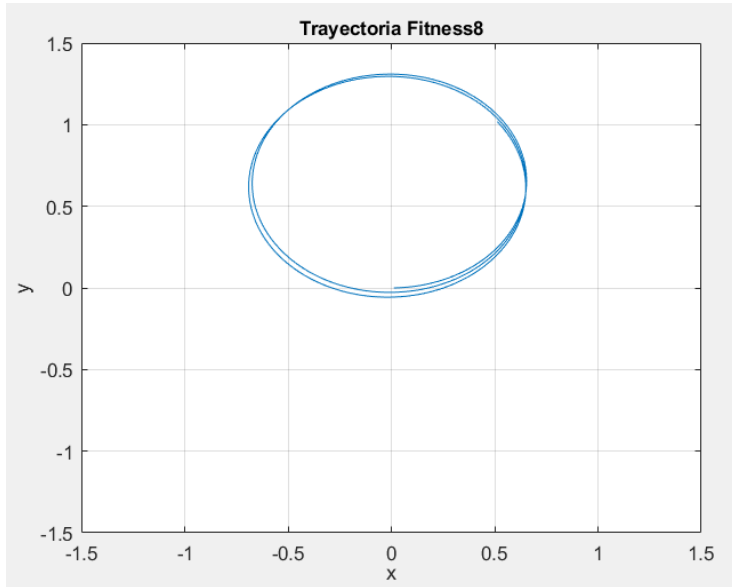


Al haberse implementado la misma expresión de fitness que en el experimento anterior, su valor es bastante similar, salvo que en este caso su crecimiento entre generaciones es algo más lineal.

En cuanto a la simulación, esta sigue sin dar un resultado óptimo, describiendo el mismo recorrido que en el experimento anterior.



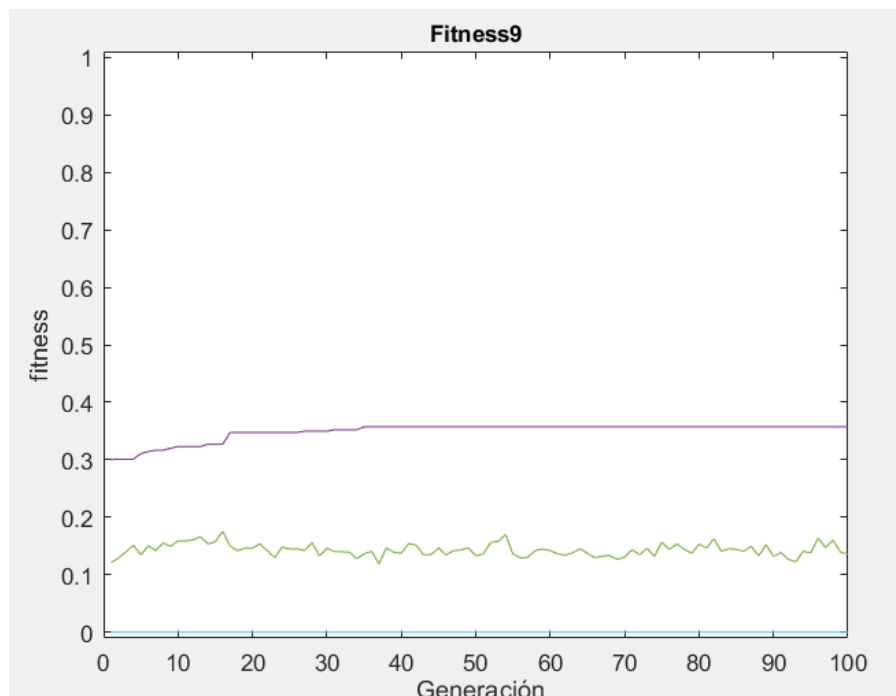
Las gráficas de la trayectoria y de la batería son prácticamente iguales a las del experimento 7.



- **Fitness 9:**

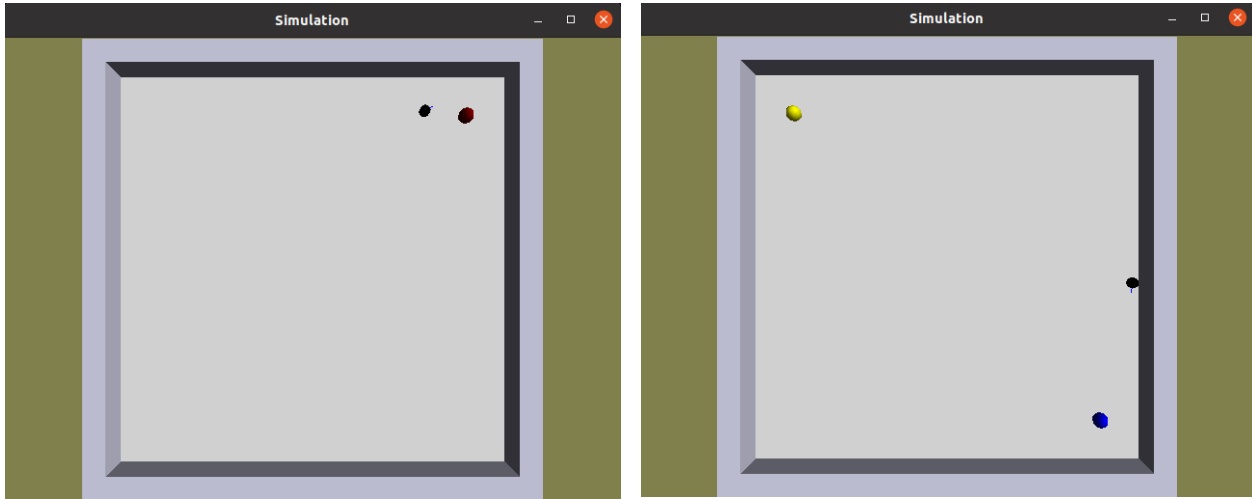
$$F = \frac{\sum_{i=0}^{N_{steps}} \left[\frac{1}{4} V(1 - \sqrt{\Delta v}) + \frac{3}{4} \left[RL_{off} BC_{on}(L_0 + L_7) \max\{L_i\} Bat + RL_{off} BC_{off}(BL_0 + BL_7)(1 - Bat) \max\{BL_i\} + RL_{on} \frac{\left(\frac{RL_7 + RL_6}{2}\right) + \max\{RL_i\}}{2} \right] \right]}{N_{steps}} \left(1 - \min\left\{ \frac{N_{coll}, 10}{10} \right\} \right)$$

Para el último experimento de la red dinámica optamos por volver a usar la red neuronal empleada en los experimentos finales de la red estática. Dado que los resultados de los experimentos anteriores no eran satisfactorios, decidimos cambiar un poco la función fitness, volviendo a los primeros experimentos donde recurríamos a la multiplicación de los distintos componentes. Debido a esto, el valor de la fitness será menor como se deduce de ver la gráfica siguiente:

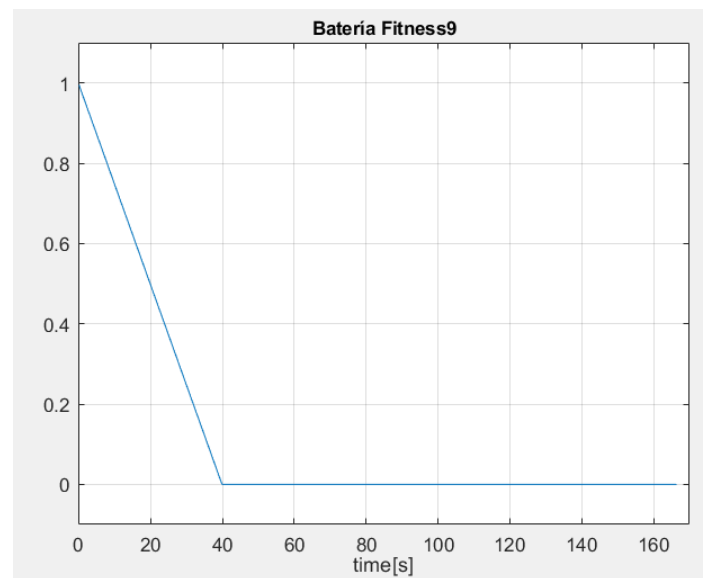
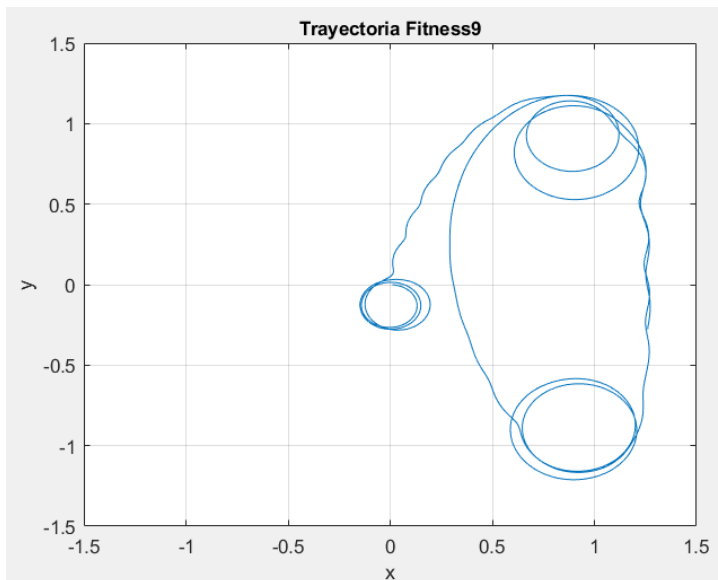


Y, al igual que ocurría en los experimentos 4 y 5, el valor 0 de la batería afecta negativamente a la función de fitness reduciendo su valor.

Ahora el robot sí que se dirige a la luz roja y da vueltas alrededor de esta. También cuando se encienden el resto de luces el robot ataca a la luz azul, nunca se dirige a la luz amarilla para cargarse.



En la gráfica de la batería se puede ver como el robot nunca se carga en este experimento, ya que nunca se dirige a esta luz. Por otro lado, viendo la gráfica de la trayectoria el robot inicialmente describe una trayectoria circular hasta que detecta la luz roja y decide ir hacia ella.



4. Problemática y futuras implementaciones

4.1 Problemática

Los principales problemas que surgieron en el desarrollo del trabajo vienen derivados de la red dinámica. Como se puede ver en los experimentos 7 y 8, y en cierta medida en el 9, el robot no realiza correctamente el recorrido que esperábamos. Esto es debido a que en estos experimentos el robot piensa que la ruta más favorable es dar vueltas en círculos mientras se carga y se descarga. Por otra parte en el experimento 9 surge otro problema distinto ya que el robot en el momento en el que su batería es menor que el umbral, este no se va a cargar a la luz amarilla, sino que este da vueltas alrededor de la luz azul.

Asimismo, observamos que en el experimento 5 tenemos un robot sobre-especializado, ya que, si cambiamos la luz roja de posición, aunque el robot sea capaz de ir hacia ella y dar vueltas, luego no es capaz de ir hacia la luz amarilla, puesto que se desorienta. Esta problemática es extensible al experimento 9.

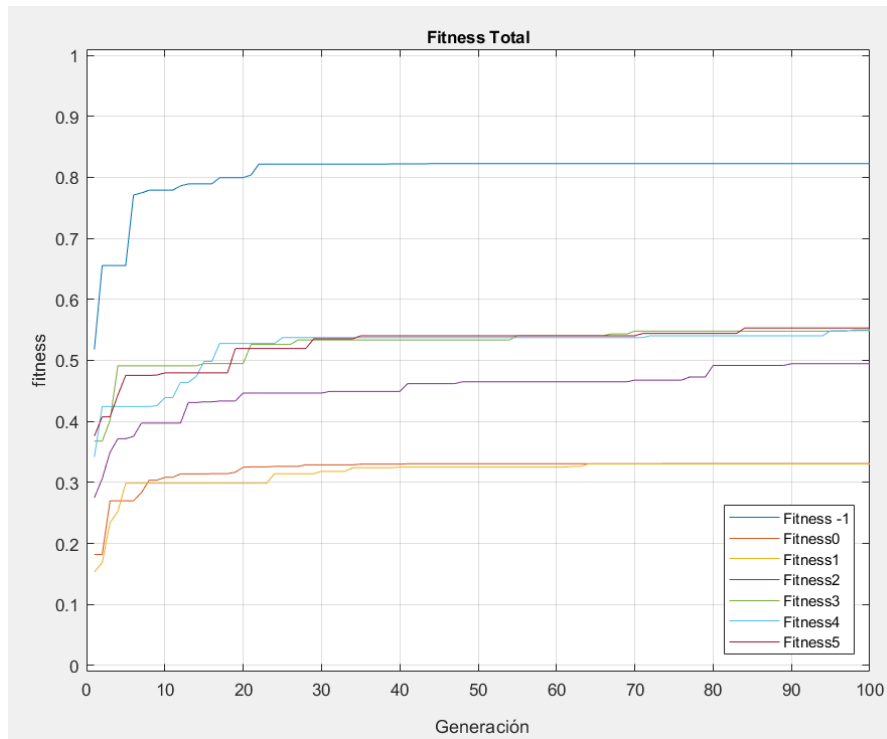
4.2 Futuras implementaciones

Se nos ocurren diversas implementaciones, de entre las cuales destacamos:

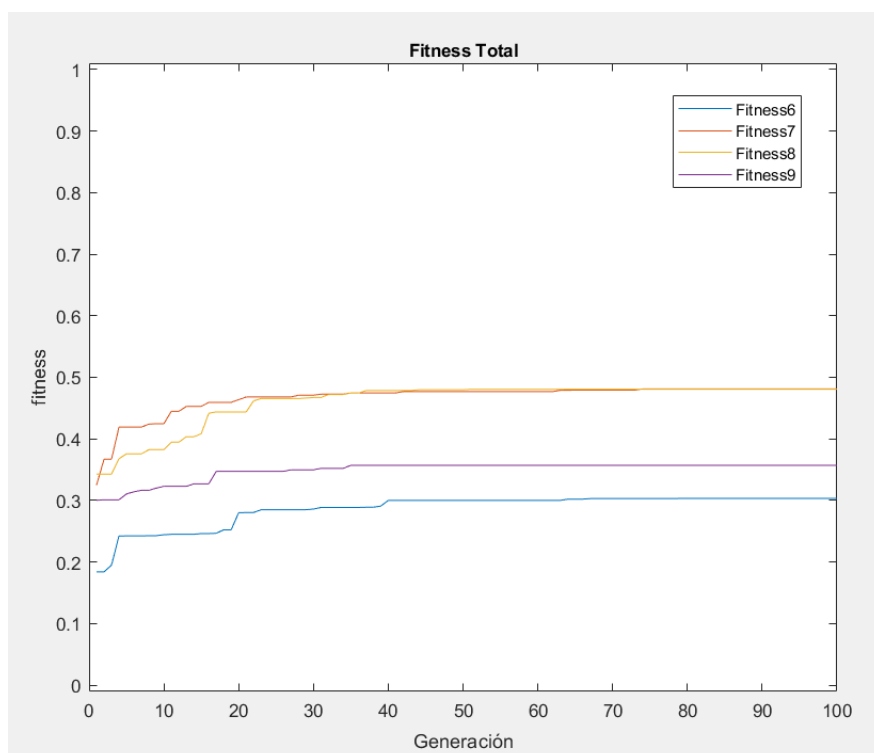
- Solución al problema de la sobre-especialización comentado previamente entrenando al robot con posiciones de luz aleatorias.
- Uso de distintas baterías, no solo centrándonos en la proveniente de la luz amarilla.
- Entrenar un robot con muchas generaciones, del orden de 3.000, implementando diversas funciones de fitness para así obtener un robot que sepa adaptarse a cambios en el entorno.

5. Conclusiones

A continuación, a modo de resumen, incluiremos una gráfica con todas las fitness de los experimentos realizados por un lado en la red estática y por otro en la dinámica.



Como hemos ido comentando, la fitness -1 (punto de partida) presenta el máximo valor, pero no es orientativo debido su poca capacidad de realizar la tarea prevista. Por otro lado, las fitness 0 y 1, tienen los valores más bajos al haberse definido como multiplicación de sensores. Viendo la 2 y 3, hay una pequeña diferencia entre ellas que reside en la máxima intensidad luminosa. La número 4, ya implementa el movimiento a la luz roja y tiene un valor muy similar a la 5, la cual permite al robot girar alrededor de la luz.



Esta nueva red, al no implementar nuevas funciones de fitness, sus valores no se alejan de los experimentos vistos en la primera parte.

Finalmente, nos hemos dado cuenta de que el empleo de arquitecturas neuronales es algo complicado y menos intuitivo si lo comparamos con el anterior trabajo, puesto que los resultados se alejaban de lo que queríamos y no permitía un cambio instantáneo del código.

Anexo. Ficheros de parámetros

- Exp -1
 - iriNeuronTemplate.txt (15% mutaciones)
- Exp 0
 - iriNeuronTemplate.txt (15% mutaciones)
- Exp 1
 - iriNeuronTemplate.txt (20% mutaciones)
- Exp 2
 - iriNeuronTemplate.txt
- Exp 3
 - iriNeuronTemplate.txt
- Exp 4
 - iriNeuron2Template.txt
 - Descomentar lightobject y bluelightobject, para para que se enciendan y se apaguen.
- Exp 5
 - iriNeuron2Template.txt
 - Descomentar lightobject y bluelightobject, para para que se enciendan y se apaguen.
- Exp 6
 - iriNeuron3Template.txt
- Exp 7
 - iriNeuron4Template.txt
- Exp 8
 - iriNeuron3Template.txt
- Exp 9
 - iriNeuron5Template.txt
 - Descomentar lightobject y bluelightobject, para para que se enciendan y se apaguen.