

UNIVERSIDAD POLITÉCNICA DE MADRID
ESCUELA TÉCNICA SUPERIOR DE INGENIEROS DE TELECOMUNICACIÓN

Trabajo O2: LAlghthouse Robot sigueluces evolutivo basado en arquitecturas neuronales

Introducción a la Robótica Inteligente 2020 - 2021

Grupo 10

Claudia Garoé Fernández García
Marcos Gómez Bracamonte

1. Introducción

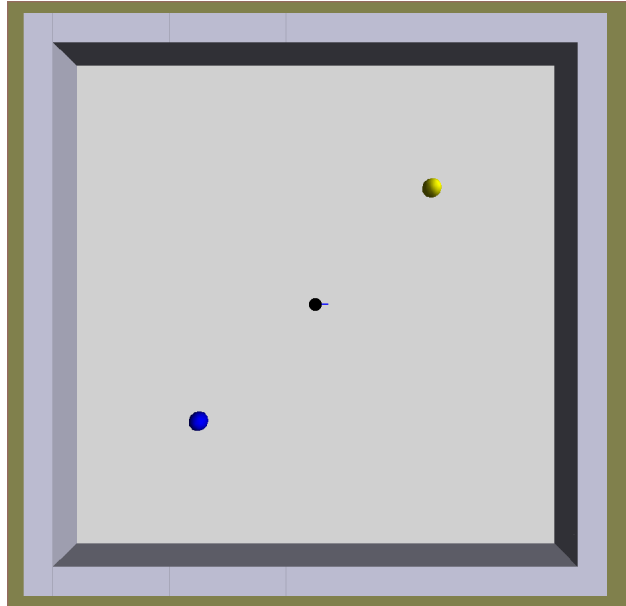


Figura 1: Entorno de la arena por defecto que se ha empleado en el entrenamiento de las distintas arquitecturas y funciones de fitness

En este trabajo abordamos el problema de la oscilación de un robot entre luces de distintos colores haciendo uso, tanto sus sensores de luz, como sus baterías.

En un primer momento, se planteó un problema basado en la oscilación entre tres luces dispuestas triangularmente, pero finalmente preferimos simplificar a un entorno con dos luces, con el propósito de poder analizar así mayor variedad de funciones de fitness y arquitecturas neuronales.

Así pues, el problema que definimos es el siguiente: el robot deberá oscilar entre dos luces, una amarilla y otra azul, describiendo una trayectoria lo más eficiente posible entre ambas y cargando a medida que se acerca a cada luz su respectiva batería.

2. Funciones de fitness desarrolladas

Para permitir al robot alcanzar la resolución del problema planteado, es necesario definir una métrica empírica para «evaluar» el desarrollo y comportamiento del robot en distintos intentos de resolver la tarea. Estos intentos se ejecutan como resultado de un proceso de evolución genética sobre cromosomas artificiales que representan la influencia de los distintos estímulos del entorno. La arquitectura de dicho procesamiento genético y su proceso de aprendizaje se detallarán en apartados sucesivos.

2.1. stdev-fitness

Esta función se basa en el cálculo de la desviación estándar del nivel de carga de las baterías azul y amarilla. Esta fitness premia que los niveles de batería sean lo más similares posibles, aunque estos niveles no son necesariamente los máximos. De esta manera, se fomenta la carga de una luz cuando está más baja que la otra.

Por otra parte penaliza mucho al robot si alguna de las baterías cae por debajo de un umbral definido de carga. Adicionalmente emplea `maxSpeedEval` y `sameDirectionEval` para influir en que el robot vaya en una misma dirección y no se quede parado con velocidad 0. Asimismo, se penaliza al robot cuando colisiona contra las paredes del entorno.

$$F_{stdev} = \frac{\sum_{i=1}^{N_{steps}} (1 - \sigma_b) \cdot p \cdot L_{goalEval} + \frac{1}{2}V(1 - \sqrt{\Delta v})}{N_{steps}} \max \left\{ 1 - \frac{\min(coll\ nr, 20)}{20} \right\} \quad (1)$$

$$\sigma_b = \sqrt{\frac{(B_{bat} - \mu_{bat})^2 + (Y_{bat} - \mu_{bat})^2}{2 - 1}}$$

$$L_{goalEval} = \begin{cases} B_{maxLight}, & B_{bat} < Y_{bat} \\ Y_{maxLight}, & B_{bat} > Y_{bat} \end{cases}$$

$$p = 0,0001, (B_{bat} || Y_{bat}) < 0,1 \quad 1 \text{ otherwise}$$

V y $\sqrt{\Delta v}$ corresponden a los coeficientes de máxima velocidad y de la diferencia de velocidad de las ruedas del robot, respectivamente. Son los mismos parámetros que se emplean en las funciones propuestas por Floreano y Nolfi [1].

Para su evaluación, se emplearán las arquitecturas ANN1 y ANN2, descritas más adelante.

2.2. schmitt-fitness

Basada en un multioscilador astable (Schmitt trigger) dependiente de la batería amarilla y luz azul.

$$F_{schmitt} = \frac{\sum_{i=1}^{N_{steps}} Y_{bat} \cdot p_1 + \frac{B_{10} + B_{17}}{2} \cdot p_2 + \frac{1}{2}V(1 - \sqrt{\Delta v})}{N_{steps}} \max \left\{ 1 - \frac{\min(coll\ nr, 20)}{20} \right\} \quad (2)$$

$$p_1, p_2 = \begin{cases} p_2 = 0; p_1 = 1, & status = 1 \\ p_2 = 1; p_1 = 0, & status = 0 \end{cases}$$

La variable *status* se define a través de la evolución del valor de la batería amarilla en el siguiente ciclo de histéresis:

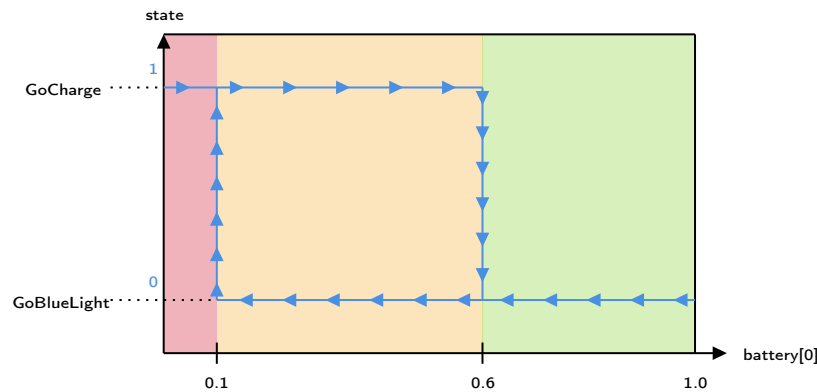


Figura 2: Diagrama del ciclo de histéresis empleado en la función de fitness schmitt-trigger

Se han definido umbrales que consideramos aceptables para la batería amarilla. En caso de que la batería amarilla tenga una carga suficiente, se favorece que el robot se dirija hacia la luz azul. Empleamos comparación estable a través del ciclo expuesto anteriormente, forzando a la fitness a estimularse exclusivamente por ciertos sensores según el estado del robot actual y pasado.

Cuando la variable vale 1, se favorece la carga; cuando vale 0, se favorece el acercamiento del robot a la luz azul. La zona roja del gráfico representa la zona donde el robot independientemente de su estado anterior se dirige a cargar la batería. Ídem para la zona verde pero en este caso para el acercamiento a la luz azul. La zona naranja representa la franja variable dependiendo del estado anterior del robot.

Para su evaluación, se emplearán las arquitecturas ANN1, ANN2, ANN3 y CTRNNosc, descritas más adelante.

3. Arquitecturas desarrolladas y análisis de comportamiento

En todas las arquitecturas neuronales se han configurado ciertos elementos comunes relacionados con su evaluación y mutación. Son los siguientes:

- **Tiempo de evaluación: 200 ticks.** Es necesario que se evalúe el tiempo suficiente en cada generación. En nuestro experimento es importante que pase el tiempo suficiente para que las baterías experimenten varios ciclos de carga y descarga, y se evalúen las situaciones pertinentes. Este es uno de los principales cambios que realizamos y que ha mejorado sustancialmente la respuesta de los robots.
- **Población: 100 individuos.** La población se ha dejado por defecto pues consideramos que es un número adecuado de individuos sobre los que entrenar cada generación.

- **Número máximo de generaciones: 10000 generaciones.** A pesar del número aquí expuesto, ninguno de los entrenamientos ha sobrepasado la generación 1000.
- **Samples por cromosoma: 2.** Es necesario más de un sample por cromosoma para permitir entrenar con posiciones de inicio aleatorias, lo que mejora en parte su respuesta frente a la sobre-especialización y discretización.
- **Crossover: Sí.**
- **Distancia de crossover: 1.** Este valor se ha mantenido por defecto para evitar variaciones muy bruscas en generaciones consecutivas
- **Tasa de mutación: 0.05 .** El valor se ha escogido para mantener una tasa de mutación no muy elevada para intentar que el aprendizaje converja sin tener muchas mutaciones en los cromosomas de generaciones consecutivas.
- **Inicio en rango aleatorio: Sí, 1.5 X, 1.5 Y.** Mejora de la robustez frente a la sobre-especialización.
- **Rango de valores de los pesos: [-5, 5].** Valor por defecto.

3.1. ANN1

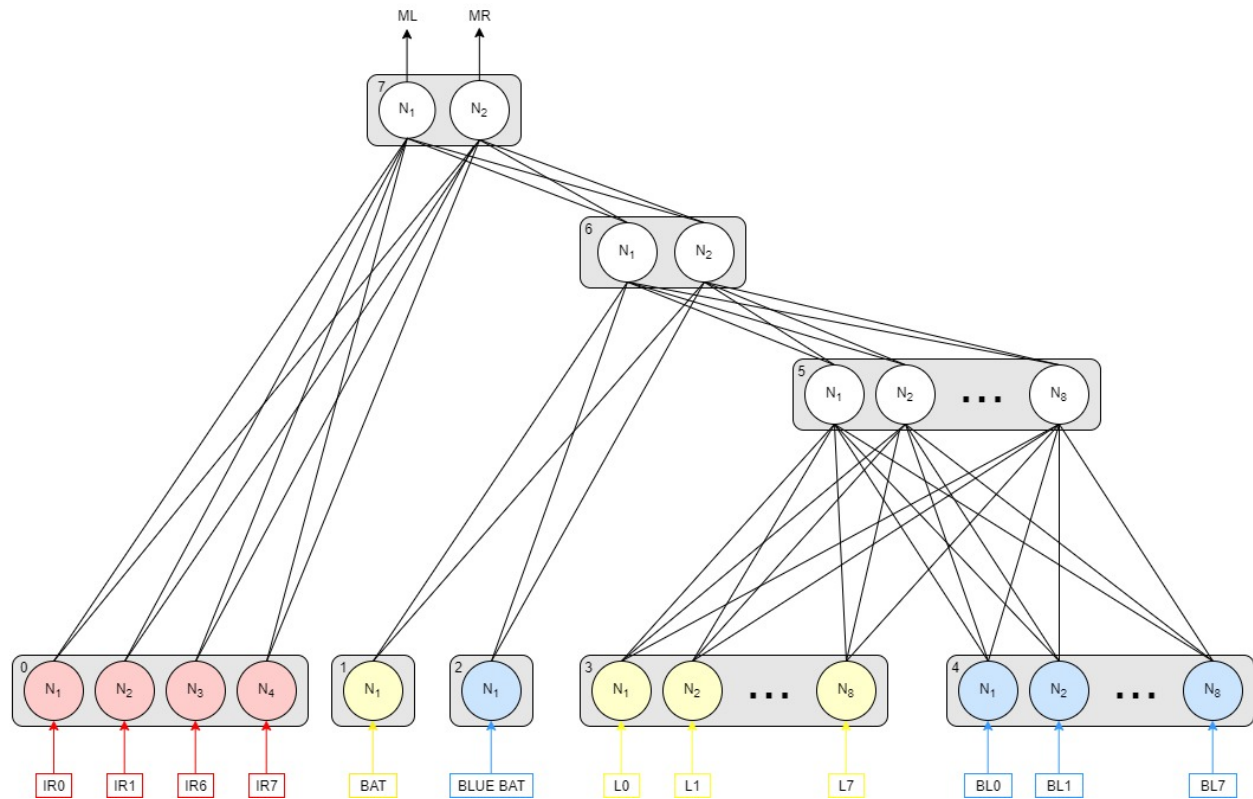


Figura 3: Primera arquitectura neuronal estática - ANN1

Como se puede observar, la arquitectura tiene ocho capas totales: la capa de sensores de proximidad por IR, que va directamente a los motores; las baterías amarilla y azul, que van a una capa intermedia (capa 6) que a su vez va a los motores; y las luces, que van a una segunda capa intermedia (capa 5) que las sintetiza antes de conectar con la otra capa intermedia (la 6) y posteriormente también a los motores.

Chromosome length: 172

3.1.1. Análisis de entrenamiento y comportamiento con stdev-fitness

Máxima fitness: 0.542584. Este resultado no es necesariamente «malo», el valor se debe a la limitación implícita del rango de la desviación estándar [0, 0.5].

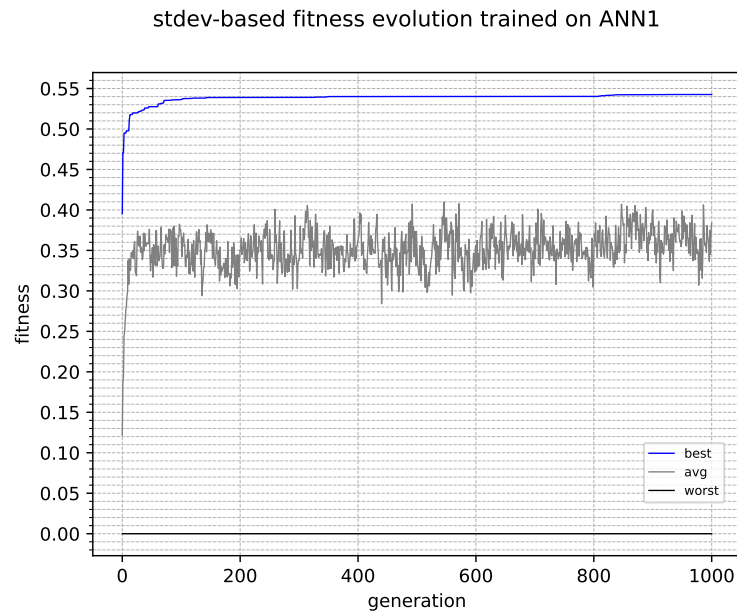


Figura 4: Evolución de la función de fitness stdev-fitness sobre la red ANN1

Comportamiento: Es aceptable, pero subóptimo. El robot va marcha atrás y realiza caminos muy largos y poco eficientes, en trayectorias cuadradas en las que se acerca a las paredes de la arena hasta que está a punto de colisionar y sólo entonces rectifica la dirección. Esto se debe a que su objetivo no es primar el nivel de carga sino minimizar la desviación entre los niveles de dichas baterías.

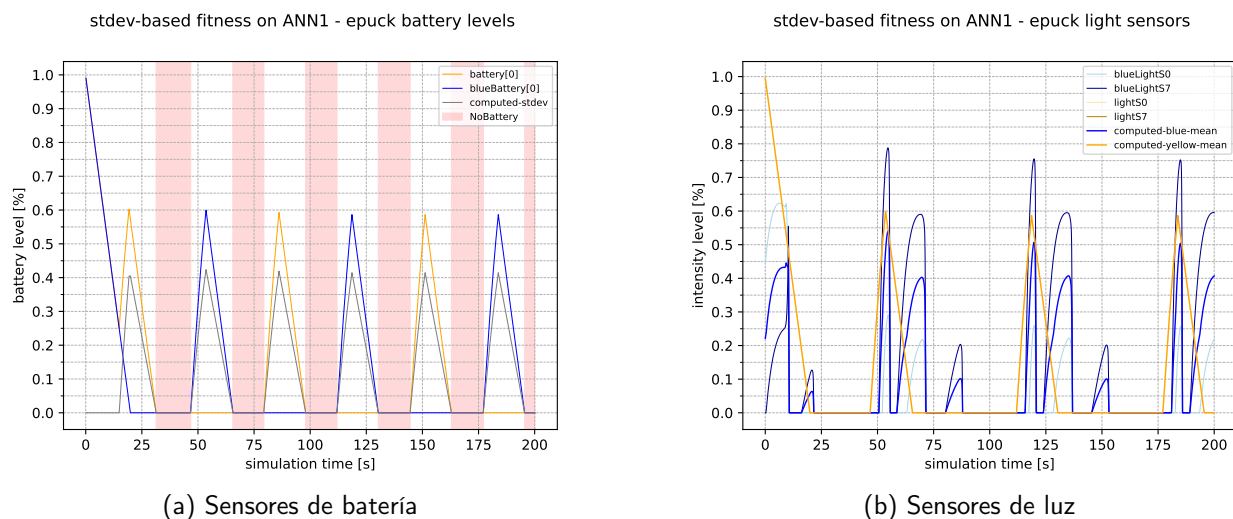


Figura 5: Evolución de los valores de distintos sensores del robot - stdev/ANN1

Se puede observar como en muchas ocasiones, durante largos periodos de tiempo, el robot se

encuentra en una situación en la que no tiene nivel de carga en ninguna batería. Como el robot minimiza la desviación estándar de los valores de carga, una situación en la que ambas baterías están descargadas es óptima.

En cuanto a los valores sensados de luz, se observan la presencia de muchos periodos de ausencia de luz, pues en estos momentos el robot está dirigiéndose hacia las paredes para rebotar y por lo tanto no visualiza ninguna luz. No obstante se puede apreciar el carácter periódico de la evolución de ambos valores.

Con respecto a la generalización, se ha probado con distintos entornos manteniendo los pesos del mejor cromosoma: al presentarse layouts compatibles con su trayecto actual, el robot realiza una ruta muy similar a aquella en el entorno en el que fue entrenado.

La evolución del comportamiento en la arena a la hora de resolver el problema se puede consultar en las gráficas de [Anexos](#).

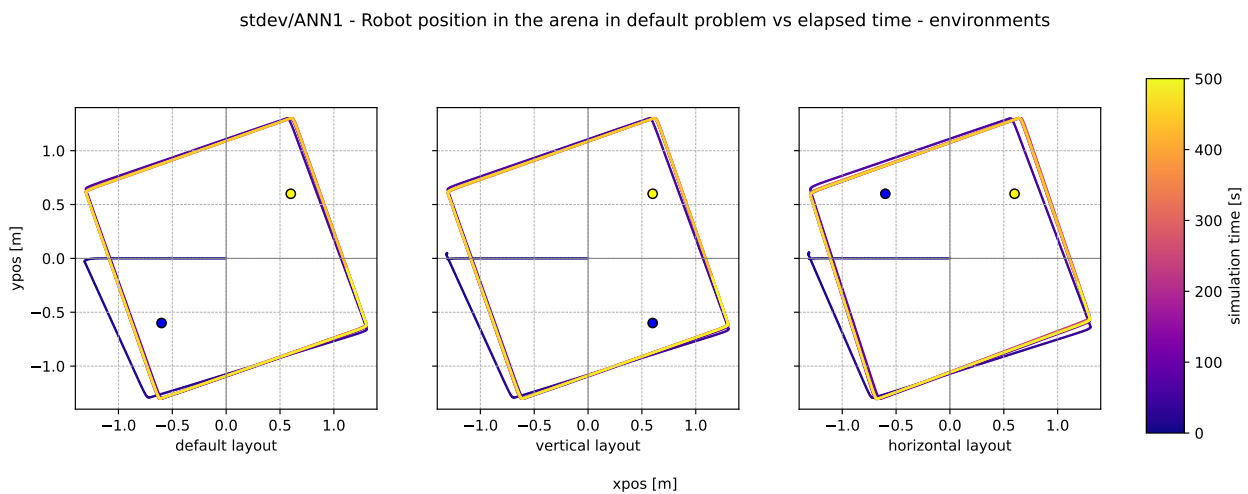


Figura 6: Capacidad de adaptación del robot en nuevos entornos - stdev-function sobre ANN1

3.1.2. Análisis de entrenamiento y comportamiento con schmitt-fitness

Máxima fitness: 0.703805, buen resultado, a continuación se expone la evolución del fitness.

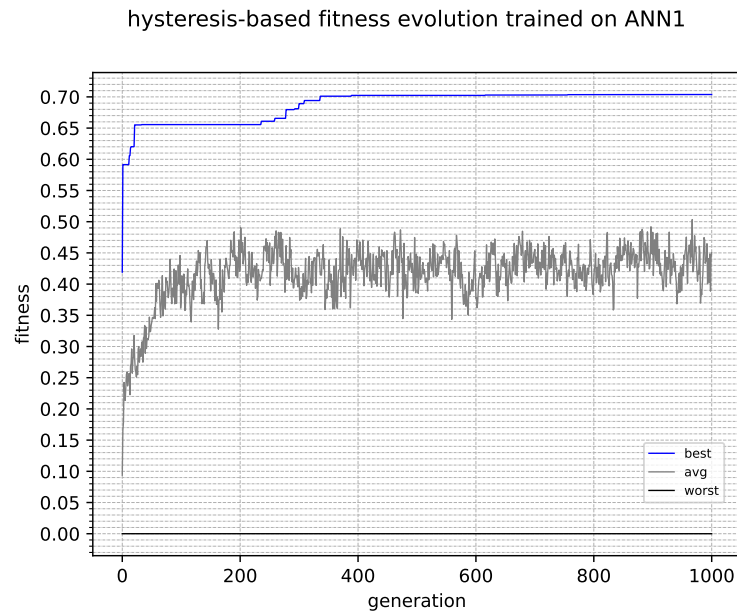


Figura 7: Evolución de la función de fitness schmitt-fitness sobre la red ANN1

La función crece mucho de repente en las primeras 20-50 generaciones para luego tener un periodo de estabilidad y volver a subir de nuevo sobre la generación 300, aproximadamente. Esta evolución se puede apreciar empíricamente en las funciones de [Anexos](#).

Comportamiento: Es bueno, realiza las oscilaciones de manera más directa que la stdev-fitness pero la trayectoria entre las luces sigue siendo abierta y tiene una forma elíptica.

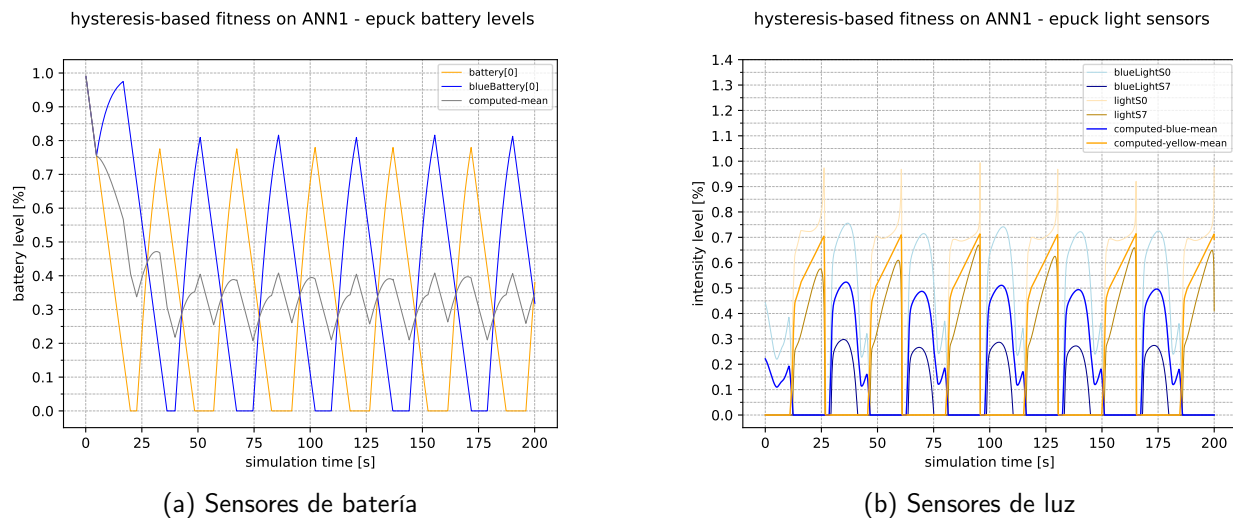


Figura 8: Evolución de los valores de distintos sensores del robot - schmitt/ANN1

Se puede observar el carácter oscilante en contrafase de los valores de las baterías azul y amarilla. Esto es debido a que ahora el robot oscila entre ambas luces mucho más rápido y de manera más directa. Esto va a causar, como se puede observar, que los niveles de media entre ambas baterías se mantengan entorno al 0.6. Aún así hay momentos en los que el robot se queda sin batería de algún color, esto se tratará de solucionar con la arquitectura ANN3.

En cuanto a los valores percibidos de luz, se aprecia como ya no aparecen periodos largos en los que el robot no visualiza ninguna luz, sin embargo los valores medios de los sensores son bajos y hay zonas donde la luz percibida de ambos colores es muy baja. Estos momentos coinciden con aquellos en el que el robot ha pasado alguna luz y está dando la vuelta para ir a la otra.

Cabe destacar que hace una generalización relativamente buena, pues al cambiar las luces de posición, el robot varía su ruta para adaptarse al nuevo entorno, aunque manteniendo trayectorias ovaladas.

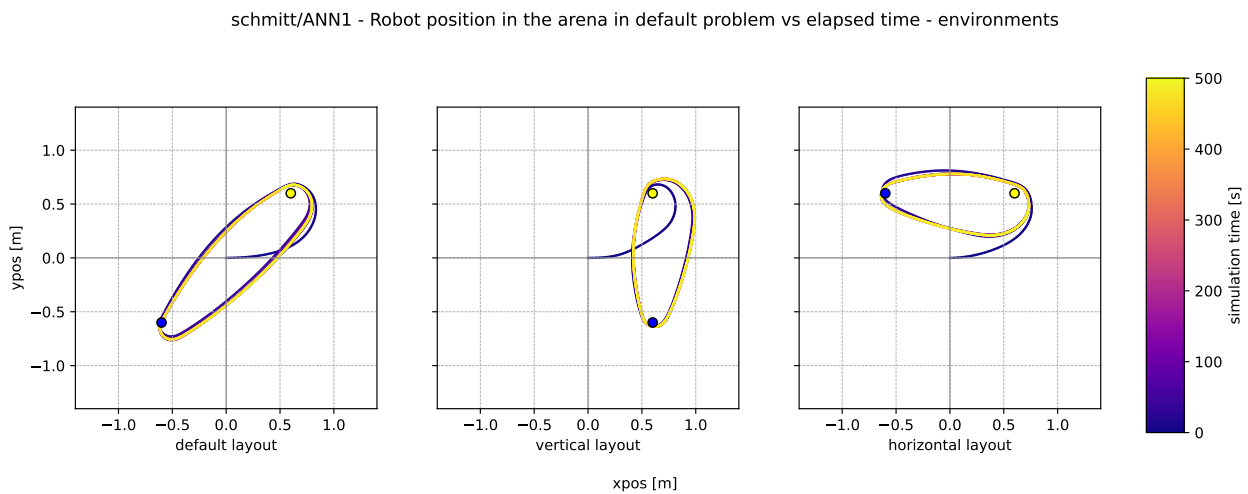


Figura 9: Capacidad de adaptación del robot en nuevos entornos - schmitt/ANN1

3.2. ANN2

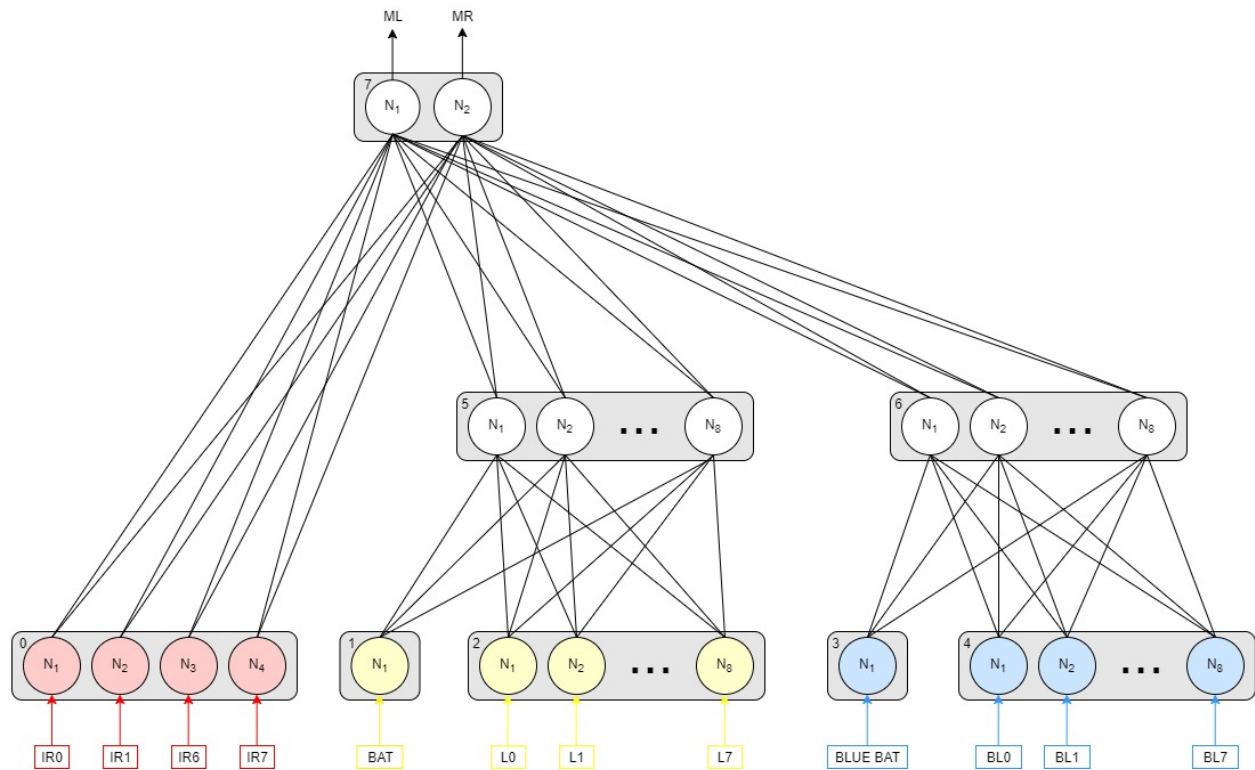


Figura 10: Segunda arquitectura neuronal estática - ANN2

Esta arquitectura reúne en dos capas intermedias los aportes de, respectivamente, batería y luz amarilla y por otra batería azul y luz azul.

Chromosome length: 202

3.2.1. Análisis de entrenamiento y comportamiento con stdev-fitness

Máxima fitness: 0.548978

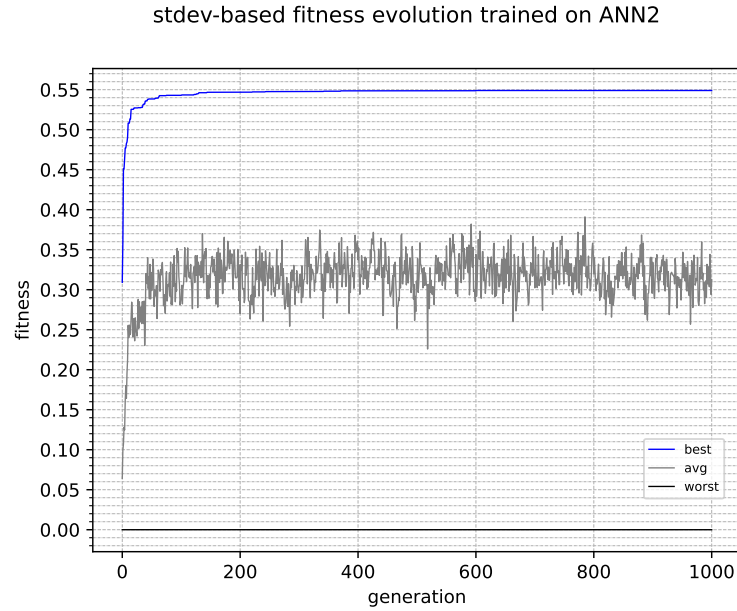


Figura 11: Evolución de la función de fitness stdev-fitness sobre la red ANN2

De nuevo, se observa un crecimiento muy similar al de la función stdev-fitness sobre ANN1. El valor máximo sigue estando limitado debido a la propia función, tal y como se ha descrito en 4.

Comportamiento: Subóptimo y similar al comportamiento sobre ANN1. Aunque ahora el robot va hacia adelante, sigue realizando un camino amplio y cuadrado, además de acercarse a las paredes tanto como puede antes de colisionar contra ellas.

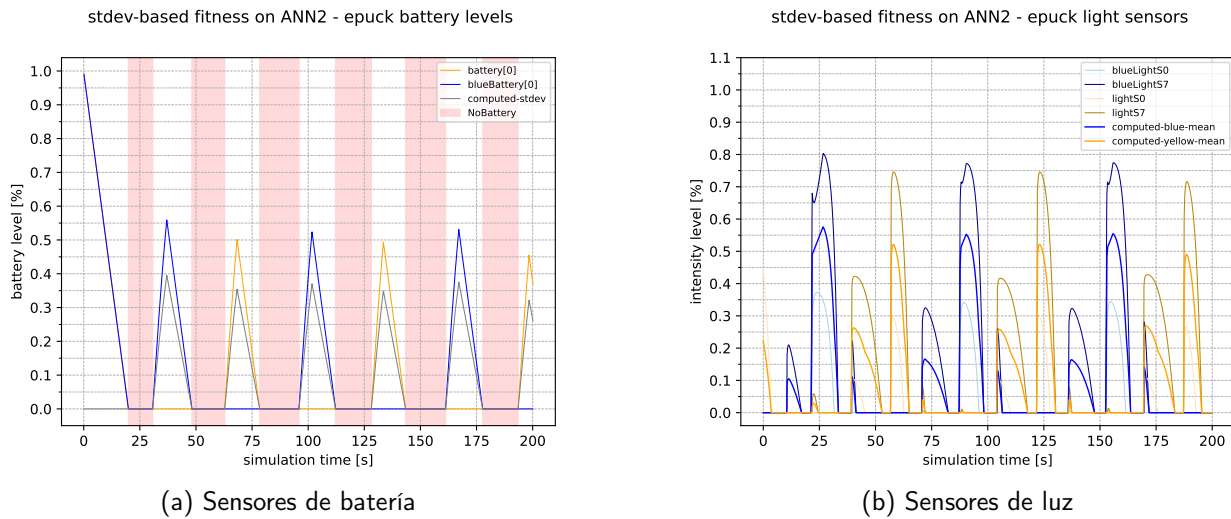


Figura 12: Evolución de los valores de distintos sensores del robot - stdev/ANN2

Se han reducido los periodos en los que no había detección de luz con respecto a ANN1, sin embargo la batería sigue cargándose y descargándose según el mismo patrón, existiendo regiones de nivel de carga 0 para ambas baterías.

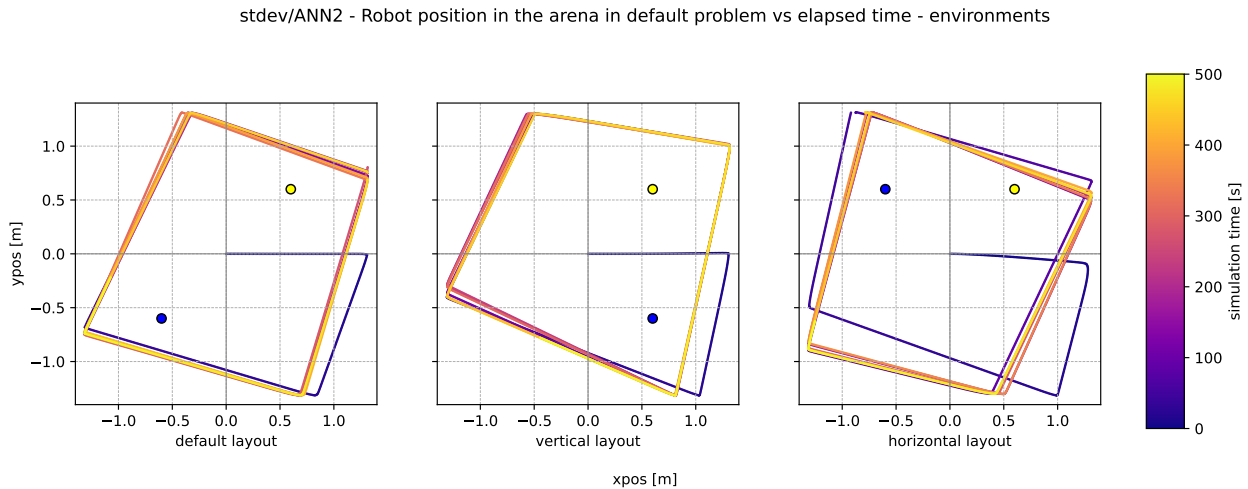


Figura 13: Capacidad de adaptación del robot en nuevos entornos - stdev/ANN2

Se observa un comportamiento de generalización muy pobre o imperceptible, debido al amplio recorrido simétrico que realiza el robot por toda la arena.

Estos resultados son similares a lo analizado en ANN1 lo que nos hizo pensar que esta función de fitness no cumpliría con el objetivo de maximizar la oscilación directa entre dos puntos de luz, por lo que su evaluación no se ha realizado en ninguna de las sucesivas arquitecturas y se ha prescindido de ella en lo que resta de experimentación.

3.2.2. Análisis de entrenamiento y comportamiento con schmitt-fitness

Máxima fitness: 0.960470. Parece un resultado prometedor...pero veremos que en realidad el comportamiento es el peor de todas las combinaciones de arquitecturas y funciones de fitness probadas.

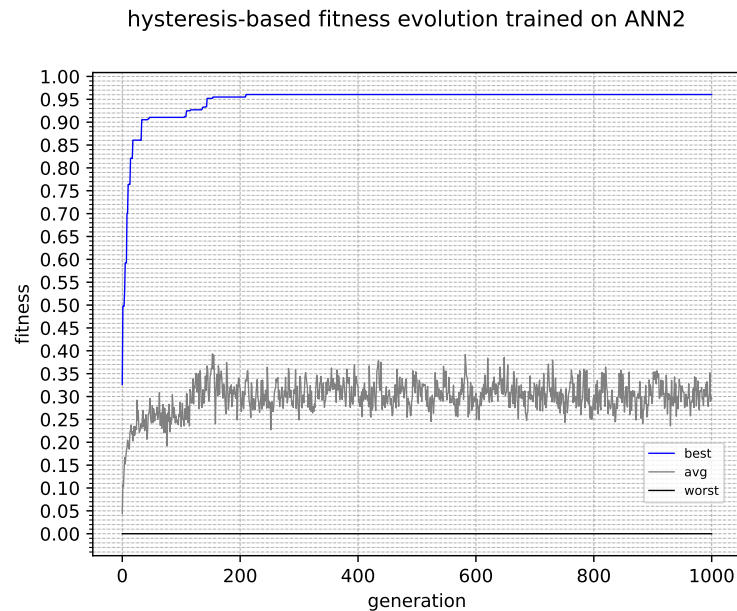
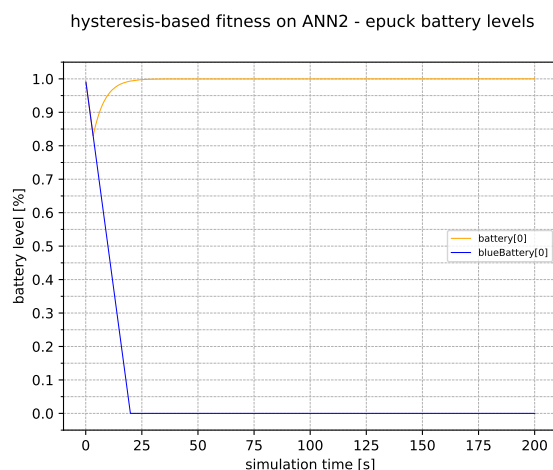


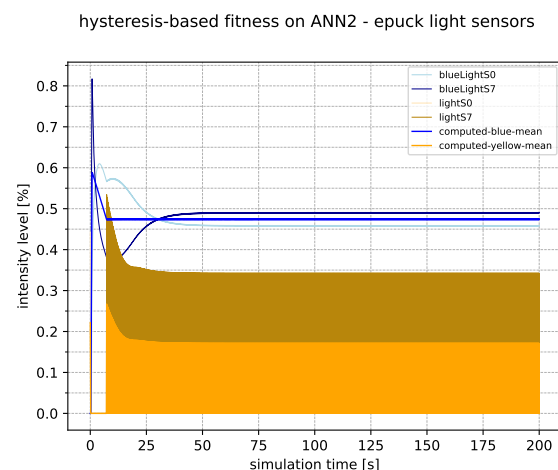
Figura 14: Evolución de la función de fitness schmitt-fitness sobre la red ANN2

A simple vista parece una evolución de entrenamiento nominal.

Comportamiento: Malo. El robot sobre especializa hasta el punto de que, después de oscilar entre las dos luces, se queda en la posición de una de ellas quieto o vibrando. Esto ocurre además desde generaciones muy tempranas.



(a) Sensores de batería



(b) Sensores de luz

Figura 15: Evolución de los valores de distintos sensores del robot - schmitt/ANN2

Como las gráficas delatan, realmente el robot se ha quedado estancado vibrando en una posición

cercana a la luz amarilla, dándola de espaldas (debido al valor constante y mayor de luz azul en los sensores delanteros). La vibración se puede percibir como lo que parece un relleno bajo el área de la gráfica de los sensores de luz.

Esto probablemente se deba a que en esta arquitectura, antes de pasar a la última capa motora con 2 neuronas, está una capa con 8 neuronas. Es muy probable que en el proceso de sintetizar toda esa información de 8 a 2 neuronas, se pierda la influencia de los inputs referidos a las baterías y el robot tenga un comportamiento errático.

Al situar al robot en nuevos entornos, curiosamente generaliza adecuadamente, pero su comportamiento sigue sin ser el esperado. En todos los casos busca la luz amarilla y se queda ahí.

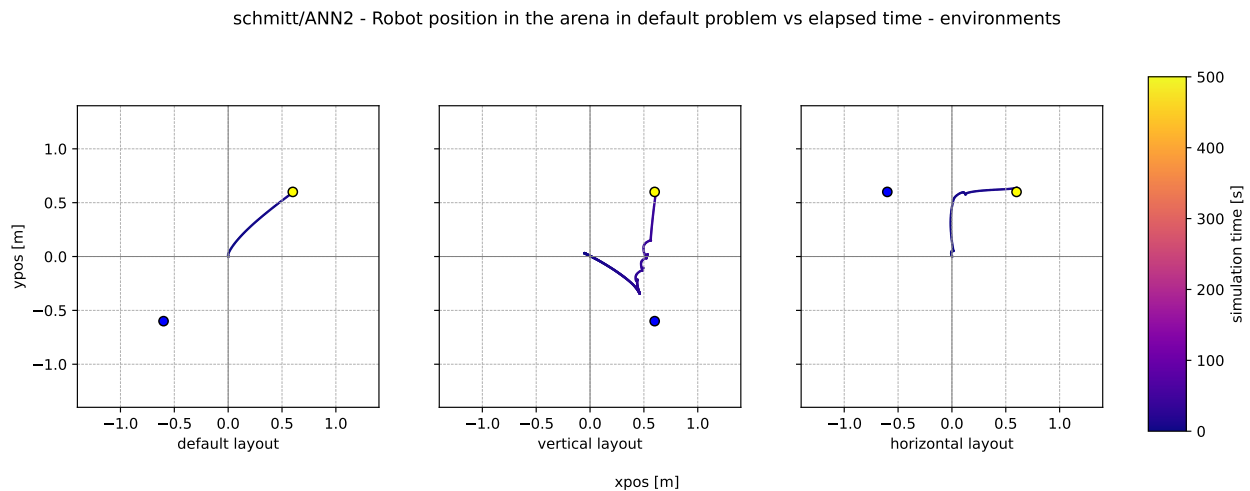


Figura 16: Capacidad de adaptación del robot en nuevos entornos - schmitt/ANN2

3.3. ANN3

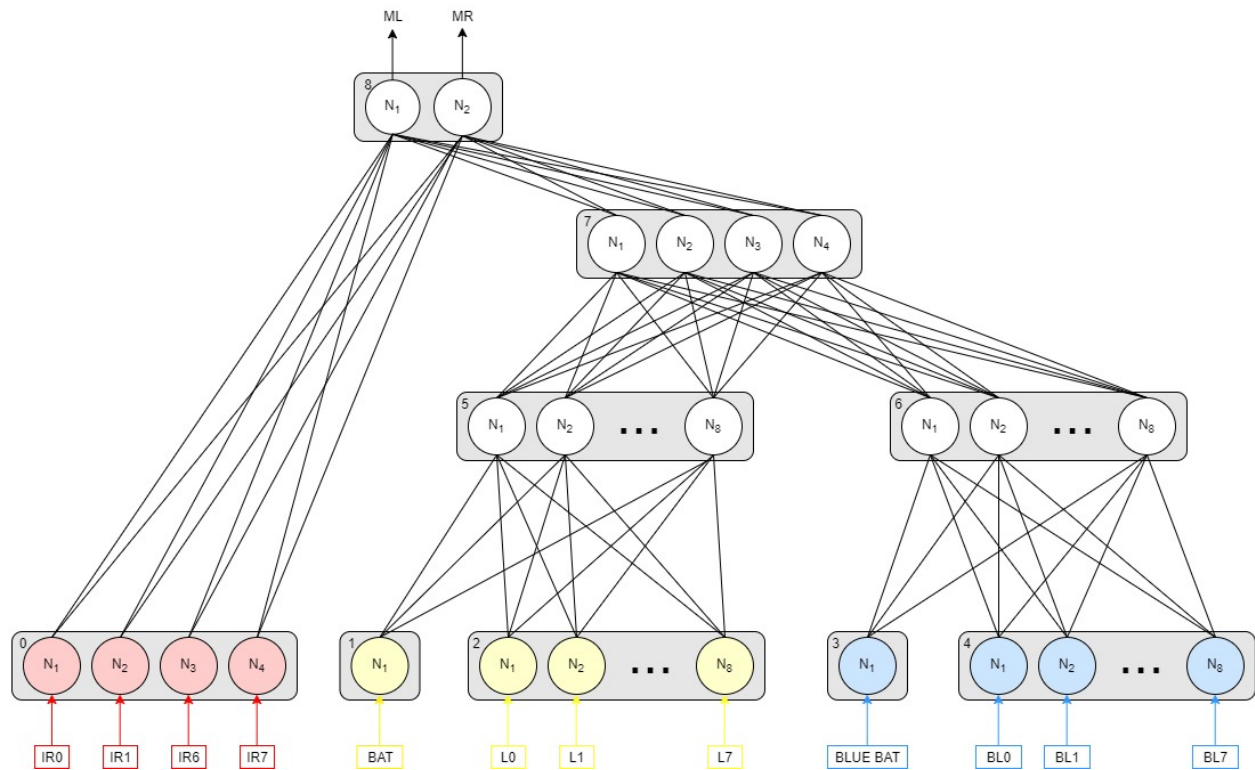


Figura 17: Tercera arquitectura neuronal estática -ANN3

Esta arquitectura es muy similar a la **ANN2**, con la diferencia de que implementa una nueva capa intermedia (la capa 7, que va a los motores), con cuatro neuronas, que sintetiza la información de las capas 5 y 6, capas intermedias que reunían la información del par de batería y luz de cada color. Esta arquitectura se ha concebido con el objetivo de eliminar el problema de la red ANN2.

Chromosome length: 146

3.3.1. Análisis de entrenamiento y comportamiento con schmitt-fitness

Máxima fitness: 0.739563, muy buen valor a priori.

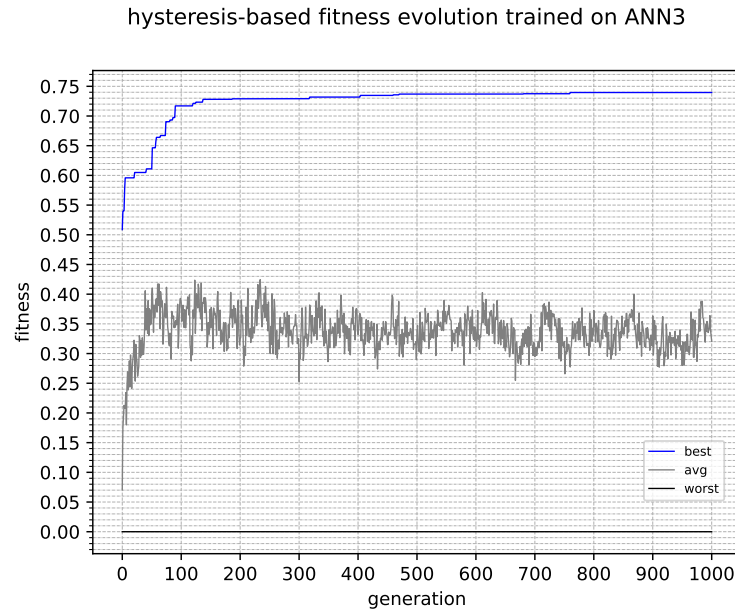


Figura 18: Evolución de la función de fitness schmitt-fitness sobre la red ANN3

Se observa un crecimiento agudo entre las generaciones 50 y 100. El impacto directo de esta subida repentina sobre el movimiento del robot se puede consultar en las gráficas en [Anexos](#).

Comportamiento: Es muy bueno, el mejor de todas las arquitecturas estáticas probadas. El robot realiza una trayectoria prácticamente directa oscilando entre las dos luces, y la velocidad del movimiento es más elevada que en las otras dos arquitecturas, haciendo que el periodo de oscilación sea menor.

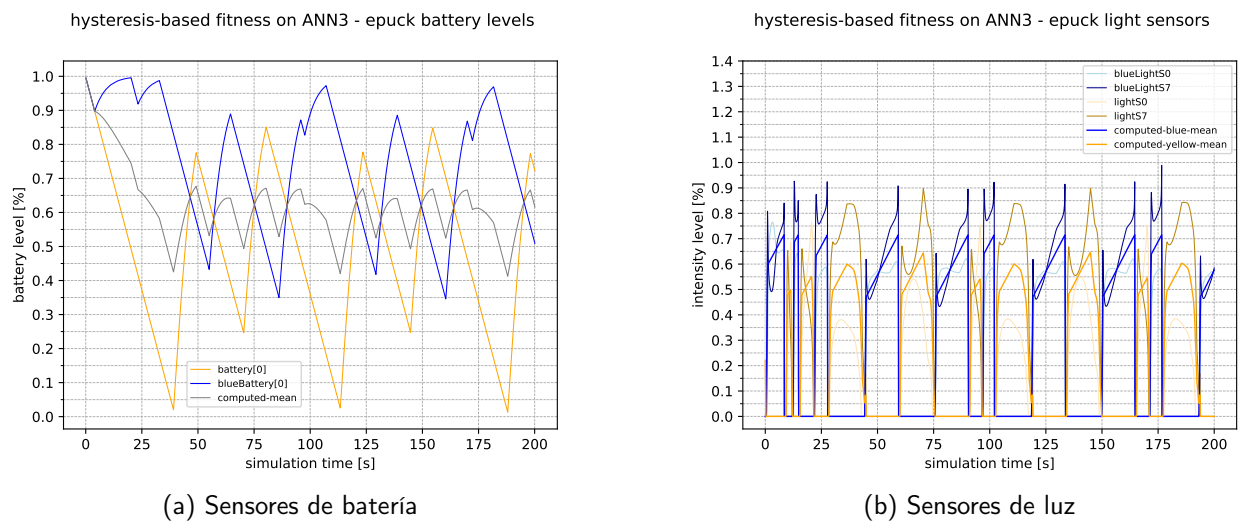


Figura 19: Evolución de los valores de distintos sensores del robot - schmitt/ANN3

Los niveles de batería nunca llegan a ser 0, y tienen una media relativamente elevada, alrededor del 0.6. Respecto a las luces, se van detectando de manera alternativa y prácticamente constante; no hay momentos en los que el robot no visualice ninguna luz, y cuando lo hace son valores altos. Obsérvese el carácter claramente oscilante en ambas gráficas.

Además, esta función entrenada sobre esta arquitectura permite al robot tener una muy buena generalización, funcionando correctamente en entornos distintos a aquel sobre el que se entrena.

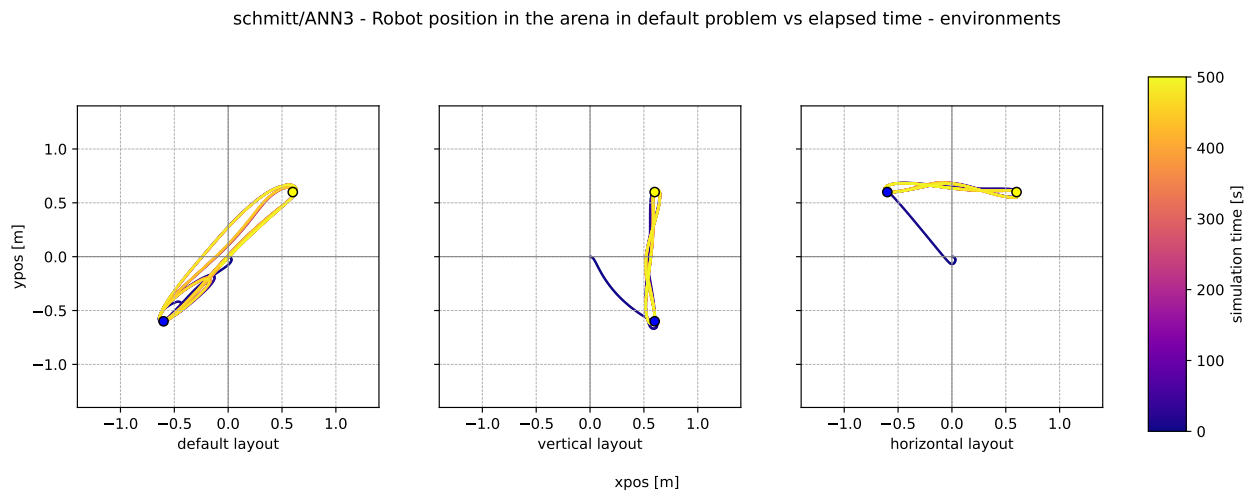


Figura 20: Capacidad de adaptación del robot en nuevos entornos - schmitt/ANN3

3.4. CTRNNosc

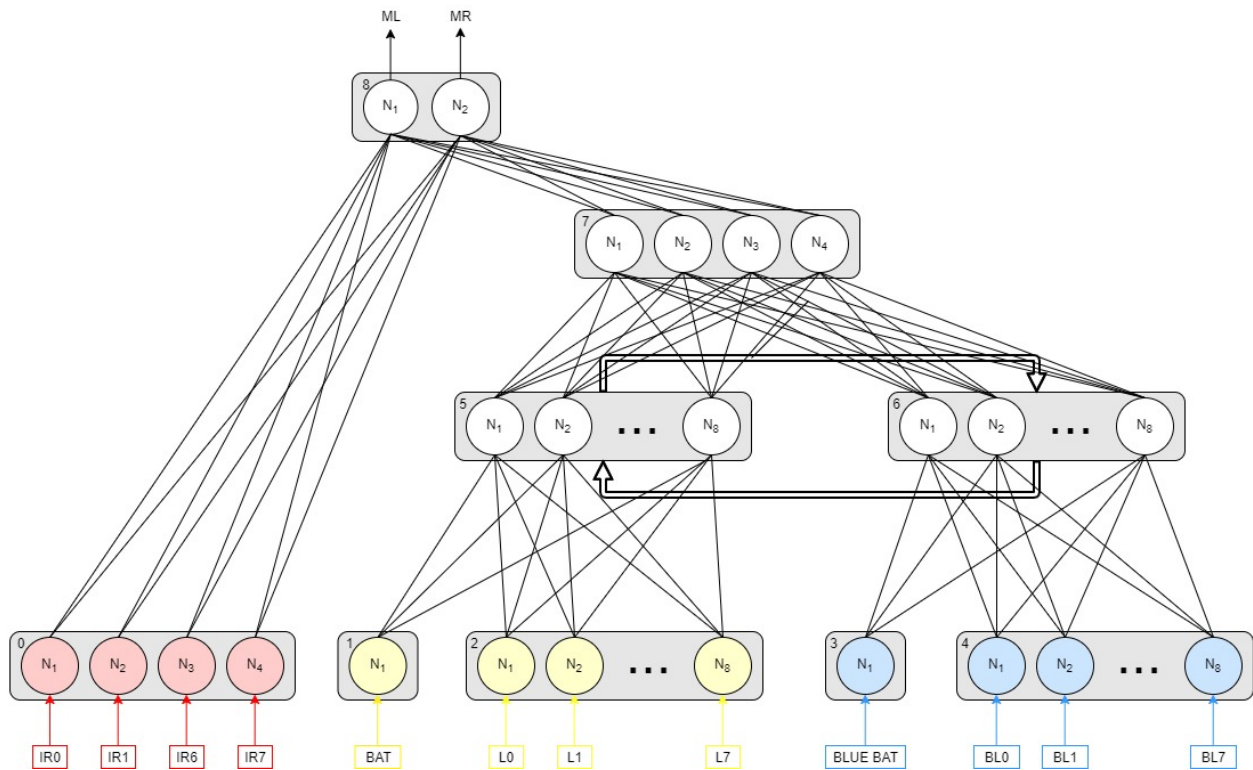


Figura 21: Arquitectura neuronal dinámica - CTRNNosc

Como podemos observar en el esquema, se trata de una arquitectura dinámica, construida a partir de ANN3, la arquitectura estática con mejor comportamiento. Se ha escogido que haya realimentación entre las capas intermedias 5 y 6 y la capa de motores. Previamente se probó realimentación en cada capa intermedia con distintas combinaciones (5, 6 y 7), pero en todas esas pruebas, el robot acababa dando vueltas alrededor de la luz amarilla, probablemente como resultado de la propia inhibición de las capas realimentadas a sí mismas.

Finalmente, se nos ocurrió realimentar las capas 5 y 6 de manera bilateral, produciendo un resultado curioso.

Chromosome length: 418

3.4.1. Análisis de entrenamiento y comportamiento con schmitt-fitness

Máxima fitness: 0.699539. A primera vista, parece un valor favorable.

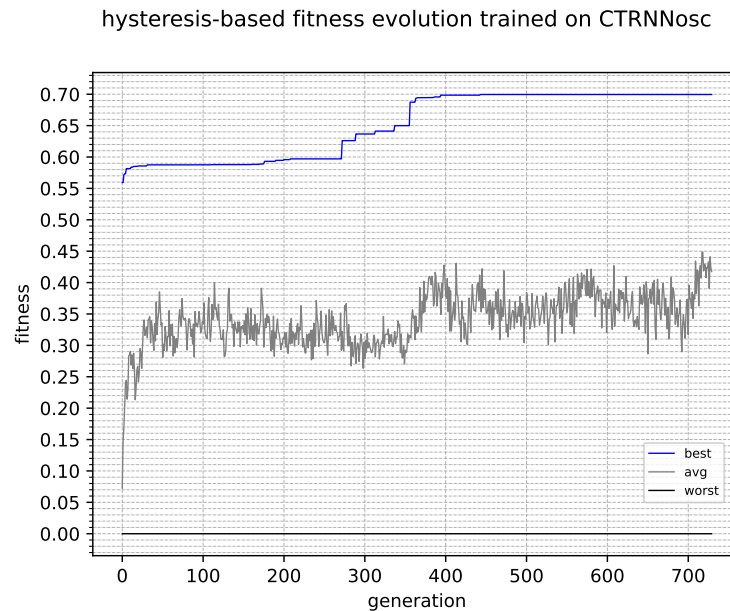


Figura 22: Evolución de la función de fitness schmitt-fitness sobre la red CTRNNosc

Se observa un crecimiento lento hasta la generación 300 aproximadamente, durante esta etapa, el robot realizaba un movimiento circular de radio la distancia del centro a las luces. A partir de esta generación se observa el comportamiento actual. Se puede observar dicha diferencia en las gráficas presentes en [Anexos](#). En este caso hubiese sido conveniente evitar la evolución a la situación actual realizando un *Early Stopping*.

El crecimiento en el fitness se produce de manera abrupta entre las generaciones 360-400.

Comportamiento: El objetivo de oscilar entre las luces lo acomete correctamente, sin embargo el robot va más rápido entre luces que sobre [ANN3](#) y se fuerza a tener que rebotar en las esquinas de la arena. Pensamos que esto es debido a que en las capas de sintetizan los niveles de luz y batería de los dos colores, al existir una conexión recíproca entre ellas, la influencia de éstas se «redobra» provocando movimientos del robot más agudos e impetuosos.

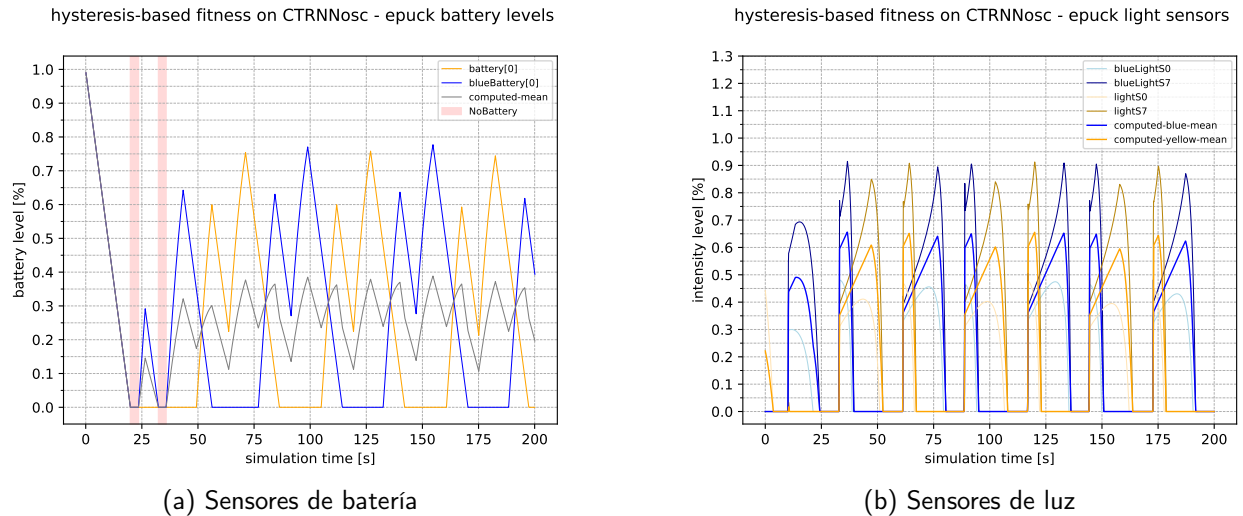


Figura 23: Evolución de los valores de distintos sensores del robot - schmitt/CTRNNosc

Se puede apreciar como al comenzar la simulación el robot tiene algunas situaciones en las que no tiene carga en ninguna de las baterías, asimismo durante gran parte de la simulación alguna batería no tiene carga. También el valor medio de carga de las baterías es algo más bajo que en el caso de ANN1 o ANN3.

En cuanto a los niveles de luz, se observan periodos cortos donde el robot no detecta frontalmente ninguna luz. Esto se debe a los instantes en los que está rebotando en las esquinas.

A nivel de generalización, esta red no es buena. Se produce una sobre-especialización que provoca que, ante cambios de entorno, el robot describa la misma ruta que aprendió cuando fue entrenado. Esto es de esperar ya que suele ser un comportamiento propio de redes recurrentes si no se trabaja en su sobre-especialización.

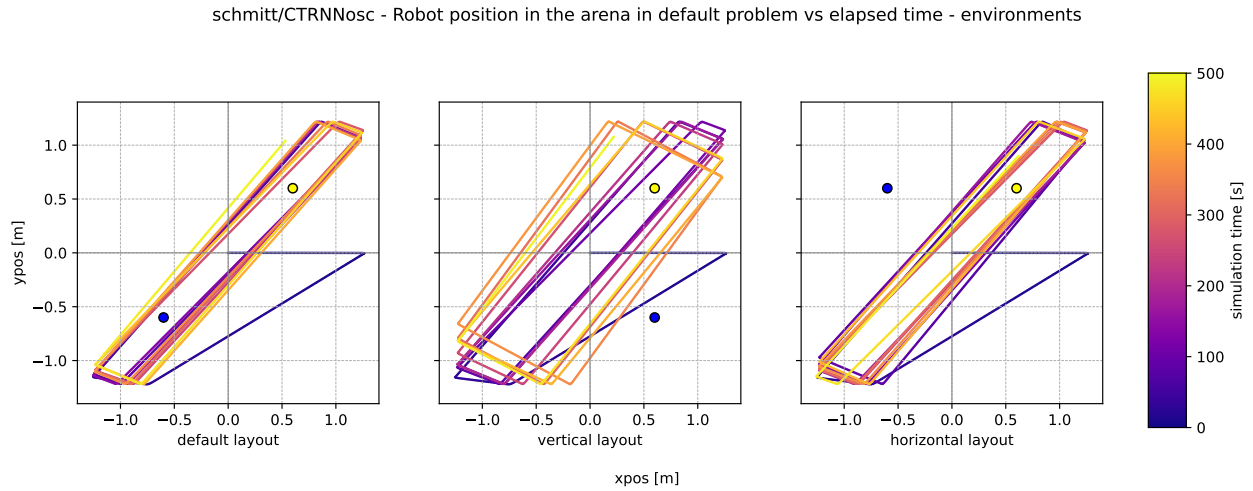


Figura 24: Capacidad de adaptación del robot en nuevos entornos - schmitt/CTRNNosc

3.5. Prueba de la función de fitness schmitt-fitness con un obstáculo intermedio añadido (CTRNNosc-obs)

Se ha probado a añadir un obstáculo intermedio en la simulación realizando un nuevo entrenamiento y evaluando su comportamiento. Se ha llegado a la conclusión de que el robot prefiere mantenerse en la luz amarilla debido a la alta penalización que supone acercarse y rebasar el muro para alcanzar la luz azul. A continuación se muestra el posicionado del robot en estas situaciones. Al no ser capaz de solventar el problema por defecto adecuadamente, la generalización es nula.

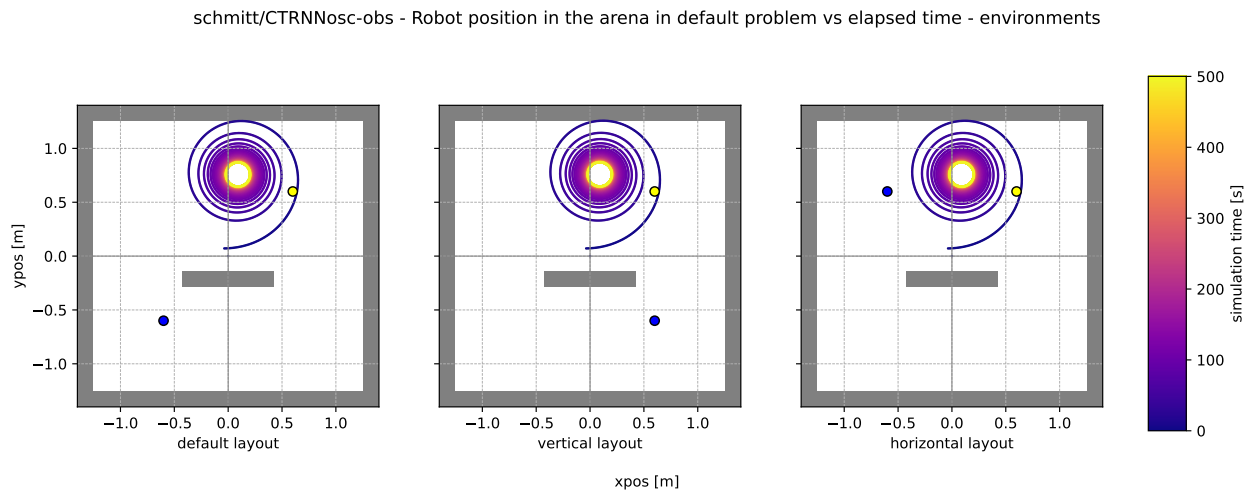


Figura 25: Capacidad de adaptación del robot en nuevos entornos - schmitt/CTRNNosc-obs

4. Anexos

- [Enlace](#) al repositorio Git del proyecto en GitHub donde se pueden encontrar los archivos completos del simulador y los archivos utilizados para realizar los gráficos para los análisis y automatizar las simulaciones, los geneticDataFiles, así como algunos vídeos sobre el comportamiento de los robots en las distintas situaciones.

4.1. Gráficas de evolución del mejor individuo en distintas generaciones en cada arquitectura con las diferentes fitness

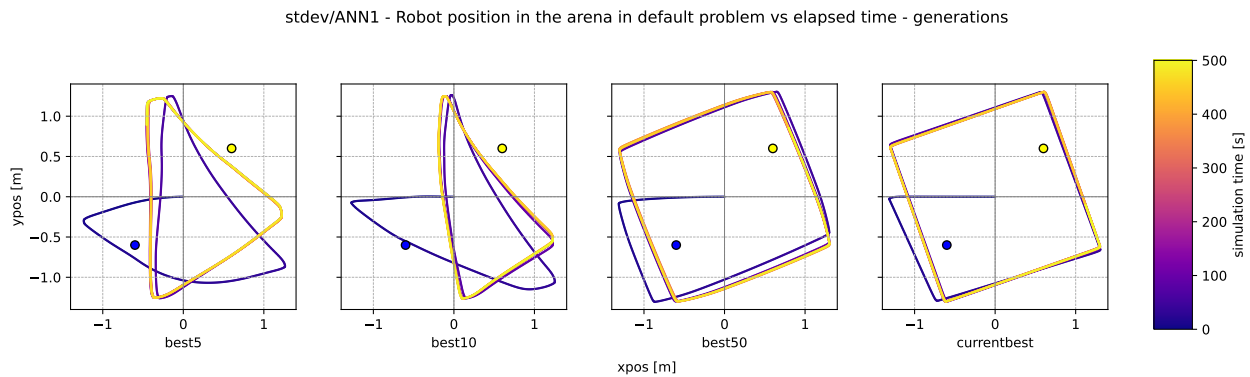


Figura 26: Evolución del comportamiento del mejor individuo de las generaciones 5, 10, 50 y el currentbest - stdev/ANN1

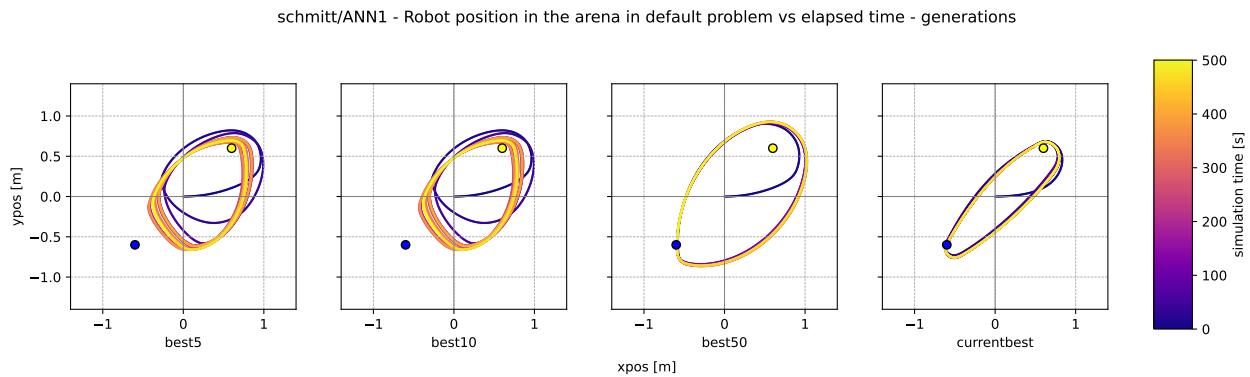


Figura 27: Evolución del comportamiento del mejor individuo de las generaciones 5, 10, 50 y el currentbest - schmitt/ANN1

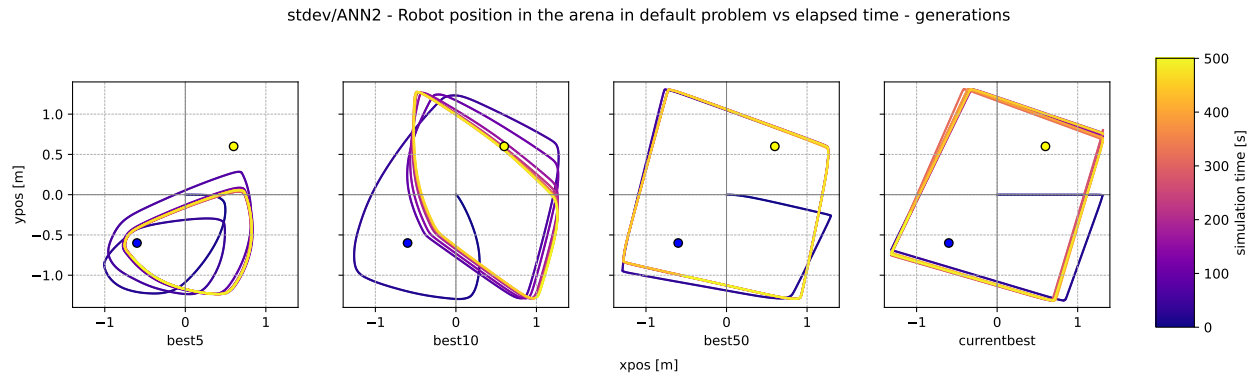


Figura 28: Evolución del comportamiento del mejor individuo de las generaciones 5, 10, 50 y el currentbest - stdev/ANN2

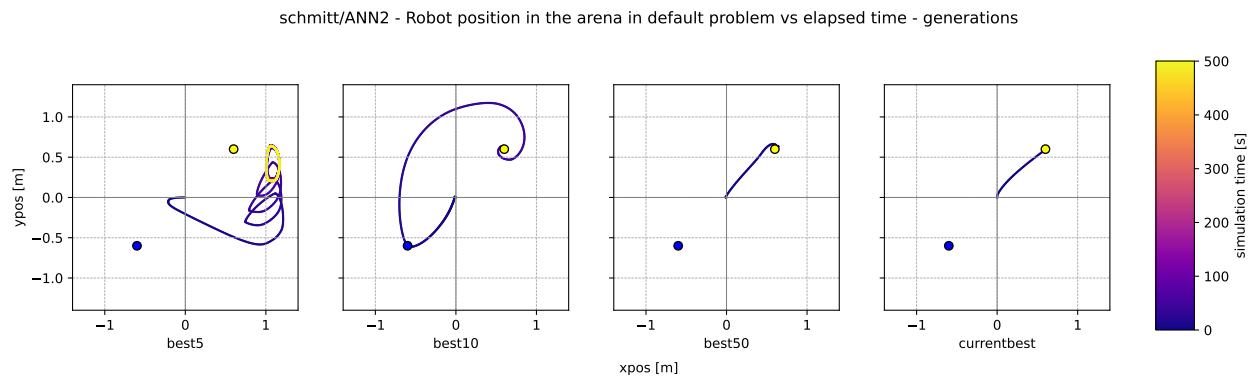


Figura 29: Evolución del comportamiento del mejor individuo de las generaciones 5, 10, 50 y el currentbest - schmitt/ANN2

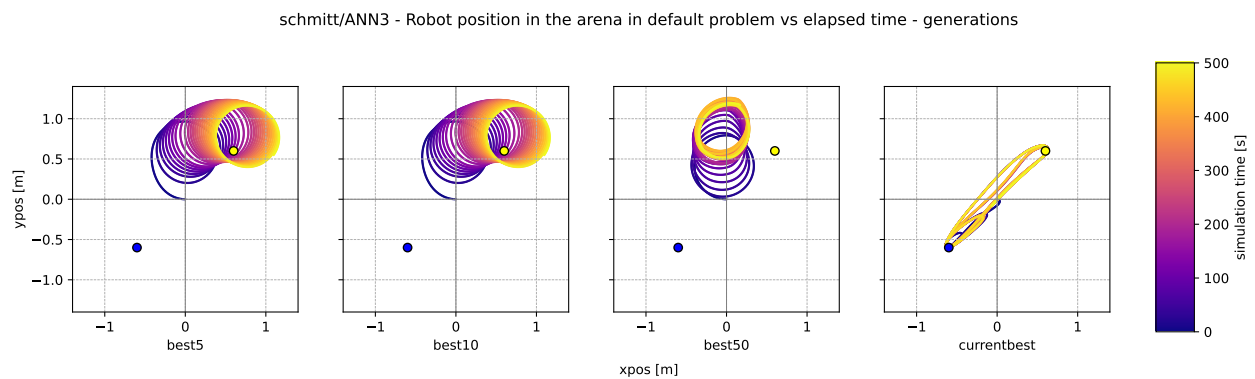


Figura 30: Evolución del comportamiento del mejor individuo de las generaciones 5, 10, 50 y el currentbest - schmitt/ANN3

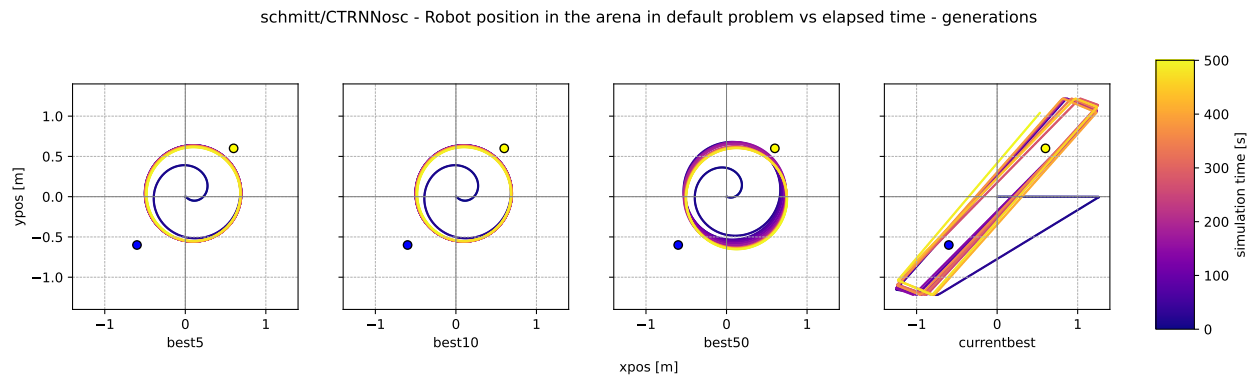


Figura 31: Evolución del comportamiento del mejor individuo de las generaciones 5, 10, 50 y el currentbest - schmitt/CTRNNosc

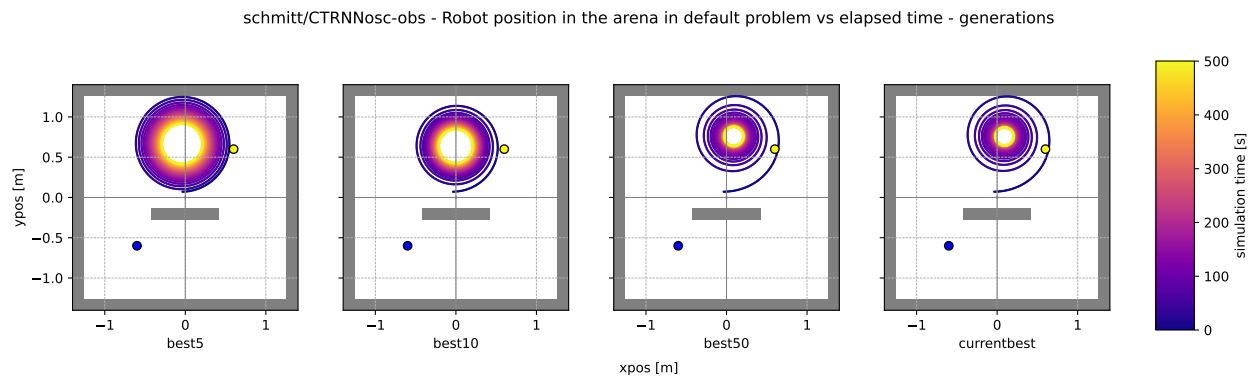


Figura 32: Evolución del comportamiento del mejor individuo de las generaciones 5, 10, 50 y el currentbest - schmitt/CTRNNosc-obs

Referencias

- [1] S. Nolfi and D. Floreano
Evolutionary Robotics.
The MIT Press, Cambridge, MA. 2000.
- [2] Felipe Mello (on Stack Exchange)
Dealing with LSTM overfitting
<https://stats.stackexchange.com/q/474012>
Stack Overflow, cross validated 2020-06-25
- [3] Álvaro Gutiérrez, Robótica Evolutiva
<http://www.robolabo.etsit.upm.es/asignaturas/irin/transparencias/ER.pdf>
Introducción a la Robótica Inteligente, abril 2021

- [4] Álvaro Gutiérrez, Genética Evolutiva
<http://www.robolabo.etsit.upm.es/asignaturas/irin/transparencias/EG.pdf>
Introducción a la Robótica Inteligente, abril 2021
- [5] Álvaro Gutiérrez, Redes Neuronales
<http://www.robolabo.etsit.upm.es/asignaturas/irin/transparencias/redesNeuronales.pdf>
Introducción a la Robótica Inteligente, mayo 2021