

Informe Starlogo TNG: Juego de Tronos (Homers)

1. Introducción

En el código adjunto se incluyen las siguientes 3 estrategias pedidas (se ejecutará una u otra según cual seleccionemos después de elegir la cantidad de tribus con el slider y de realizar el Setup de los Homers/Donuts):

1. **Nurture:** La tribu de Homers con esta estrategia estará siempre buscando Donuts de su color y yendo hacia ellos para comérselos.
2. **Destroy:** La tribu de Homers con esta estrategia estará siempre buscando Homers de otro color y yendo hacia ellos para luchar.
3. **Performance:** Esta es nuestra estrategia personalizada, que explicaré en detalle más adelante.

La estrategia seleccionada será la que el primer equipo de Homers (los Homers que se ven normal y comen Donuts verdes), llamados “**Homers 2.0**”, seguirá en su ejecución. Mientras que los demás equipos de Homers (cada uno con su color respectivo, tanto el Homer como el Donut) harán un movimiento estándar aleatorio.

Esto nos servirá para poder valorar fácilmente si la estrategia elegida es buena o no, al tener equipos “simples” en contra.

2. Implementación y funcionamiento

2.1. Nurture

Lo primero que hace **Homer 2.0** en cada bucle de esta estrategia es llamar a la Procedure “**search_Donuts_Simple**” que comprueba si hay algún agente Donut en un rango determinado alrededor suyo y, en caso afirmativo, me guardo la ID de ese agente en una variable.

Una vez tengo la ID del agente, compruebo si el Donut es del mismo color que el Homer y, en caso afirmativo, giro el Homer en dirección al Donut y llamo a la Procedure “**goTo_Target**” que avanza un paso hacia esa dirección. En caso contrario, llamo a la Procedure “**go_Forward**” que avanza un paso hacia delante y gira un número aleatorio de grados.

Llegado a este punto, vuelve a empezar el bucle.

2.2. Destroy

En esta estrategia, lo primero que hace **Homer 2.0** en cada bucle es llamar a la Procedure “**search_Homers**” que comprueba si hay algún agente Homer en un rango determinado alrededor suyo y, en caso afirmativo, me guardo la ID de ese agente en una variable.

Una vez tengo la ID, compruebo si el Homer enemigo es de diferente color que nuestro Homer y, en caso afirmativo, giro el Homer en dirección al Homer enemigo y llamo a la Procedure “**goTo_Target**” que avanza un paso hacia esa dirección. En caso contrario, llamo a la Procedure “**go_Forward**” que avanza un paso hacia delante y gira un número random de grados.

Llegado a este punto, vuelve a empezar el bucle.

2.3. Performance

Esta estrategia se basa en una **combinación** de la estrategia de sólo **buscar Donuts** y la de sólo **buscar Homers enemigos**. Además, le añadimos una función de **esquivar Donuts enemigos** (procedure “**dodge_it**”) mientras está en modo de búsqueda de comida. Nuestro equipo de Homers 2.0 va cambiando entre la estrategia de buscar comida, la de buscar Homers enemigos y quedarse quieto **según la energía que tenga** en ese momento. Esto lo controlaremos con el sistema de niveles de energía que explico a continuación con un ejemplo de ejecución:

- De inicio, Homer tendrá más energía que el mínimo pautado para moverse (**min_toAction = 102**). Además, la variable que indica si entrar en modo búsqueda de enemigos (fight) está en False por defecto, por lo que llamaré a la función que inicia la búsqueda de comida (search_Donuts). Y, al volver de esa búsqueda, en caso de que supere el nivel de energía para pasar al modo búsqueda de enemigos (**top_energy = 600**), pondré la variable fight en True para el siguiente bucle.
- Cuando Homer supere los 600 de energy, y tenga la variable fight en True, entonces iniciará la búsqueda de Homers enemigos (search_Homers). Si llega un punto que la energía baje de un mínimo (**min_energy = 300**), se volverá a poner en False la variable fight para que la siguiente vez Homer vuelva a ponerse en modo búsqueda de comida.
- Por último, si llega un momento que la energía de Homer baja del min_toAction, Homer se quedará parado para no morir.

En resumen, Homer está en un bucle que buscará comida hasta que llegue a tener 600 de energía, luego se pondrá a buscar enemigos hasta que baje a 300 de vida, volverá a ponerse a buscar comida hasta llegar a 600, etc.. Y, en caso de que baje a 102 de vida (ponemos 102 para que tengan que atacarle 3 veces para morirse), se quedará quieto.

3. Conclusiones

Adjunto la tabla de pruebas de los resultados del equipo de Homers con cada estrategia contra 3 equipos de enemigos simples:

	Nurture	Destroy	Performance
1- ¿Victoria? / ¿Tiempo?	No / 220	No / 70	No / 250
2- ¿Victoria? / ¿Tiempo?	No / 175	Si / 165	Si / 650
3- ¿Victoria? / ¿Tiempo?	No / 160	No / 30	Si / 600
4- ¿Victoria? / ¿Tiempo?	No / 145	No / 45	Si / 550
5- ¿Victoria? / ¿Tiempo?	No / 330	No / 30	Si / 450
6- ¿Victoria? / ¿Tiempo?	No / 130	No / 45	Si / 800
7- ¿Victoria? / ¿Tiempo?	No / 125	Si / 145	Si / 725
8- ¿Victoria? / ¿Tiempo?	No / 135	No / 100	Si / 600
9- ¿Victoria? / ¿Tiempo?	No / 200	No / 90	No / 750
10- ¿Victoria? / ¿Tiempo?	No / 150	Si / 135	Si / 700
%Victorias / Tiempo Medio	0% / 177	30% / 85.5	80% / 607.5

Como podemos ver en la tabla, la estrategia de siempre buscar Donuts para hacerse más fuerte no funciona porque, aunque los Homers acumulen mucha energía, llega un momento que no quedan más Donuts pero los Homers siguen buscando Donuts lo que hace que para lo único que sirva es para atrasar su muerte.

Por otro lado, la estrategia de siempre buscar Homers enemigos tiene algo más de éxito que la anterior, aunque suelen darse 2 casos: o mueren todos los Homers con esta estrategia muy rápido y pierden, o ganan la partida más o menos rápidamente. Aunque, como se puede ver, tampoco tiene mucho porcentaje de éxito.

Por último, nuestra estrategia consigue un buen ratio de victorias aunque tarda bastante en llegar a ese punto, ya que se basa en que llega un punto que nuestros Homers ya no tienen más Donuts para sacar comida y se quedan quietos, y simplemente esperan a que los otros Homers enemigos que queden vayan muriendo al también quedarse sin Donuts y no parar de moverse.