

---

# **El mundo de Wumpus**

## **(Extensión)**

Inteligencia Artificial  
Universidad de Las Palmas de Gran Canaria

Fecha: 04 de marzo de 2.003

---

**Javier Honorio Casado Fernández**

jhcasado[AT]gmail[DOT]com

**José Ángel Montelongo Reyes**

ja.montelongo[AT]gmail[DOT]com

---

### **Resumen**

En el siguiente documento se detallan las extensiones que han sido implementadas para resolver el problema del *Wumpus* y del *BuscaTesoros*. En esta nueva versión se han implementado dos nuevas estrategias para abandonar la cueva, por parte del agente *BuscaTesoros*: *DistanciaMinima* y *DistanciaMinimaA*, esta última basada en el algoritmo A\*. También se han aumentado las estrategias que puede adoptar el *Wumpus*: *Territorial* y *Asesino*, y se han añadido 12 mapas para probar las mejoras.

---

## 1 Estrategias del Agente

En esta nueva versión se han añadido dos nuevas estrategias cuando el agente tiene como objetivo abandonar la cueva. Con estas dos nuevas estrategias, el agente puede elegir entre tres estrategias posibles:

- Distancia Mínima
- Distancia Mínima usando A\*
- Aleatorio

Para cambiar la estrategia que va a utilizar el agente debemos cambiar el atributo método *metodoSalida* como se ilustra en el siguiente ejemplo:

```
(BuscaTesoros  
  (Nombre C3PO)  
  (Forma R)  
  (MetodoSalida Aleatorio)  
  (Posicion (0 1)))
```

### 1.1 Distancia Mínima

Este método se basa en abandonar la cueva avanzando a la casilla que está situada a menor distancia de una de las salidas que ha encontrado el agente durante la fase de exploración. Este método no es determinista ya que puede provocar que el agente se quede indefinidamente en la cueva dando giros. Esto se produce porque el agente no es capaz de esquivar los diferentes obstáculos. Este problema queda de manifiesto en el *Mapa05.lsp*.

### 1.2 Distancia Mínima A\*

Este otro método se basa en abandonar la cueva calculando previamente cual es la ruta óptima. Para llevar a cabo su cometido se ha utilizado el algoritmo A\* adaptándolo al problema actual. También hay que añadir que debemos de tener activada la opción que nos permite rellenar el mapa a medida que el agente vaya explorando la cueva. Este algoritmo es con diferencia el que mejores resultados produce. Toda su implementación ha sido escrita dentro del fichero *ruta.lsp*.

## 2 Estrategias del Wumpus

El monstruo posee ahora dos nuevas estrategias para moverse por la cueva. Dos de ellas

son bastante simples y no requieren obtener información acerca de donde está situado el agente, sólo la estrategia *Asesino* requiere obtener información acerca de donde está situado el agente. Las posibles opciones que posee el atributo *instinto* son:

- Aleatorio
- Territorial
- Asesino

### 2.1 Territorial

Esta estrategia es igual que la estrategia *Aleatoria* con la diferencia de que en este caso el agente se moverá aleatoriamente alrededor de su posición inicial. Existe una variable denominada *radioTerritorio* que indica cual es el radio del territorio del monstruo. Este método resulta útil si queremos que el monstruo custodie le tesoro y observar como actúa el agente.

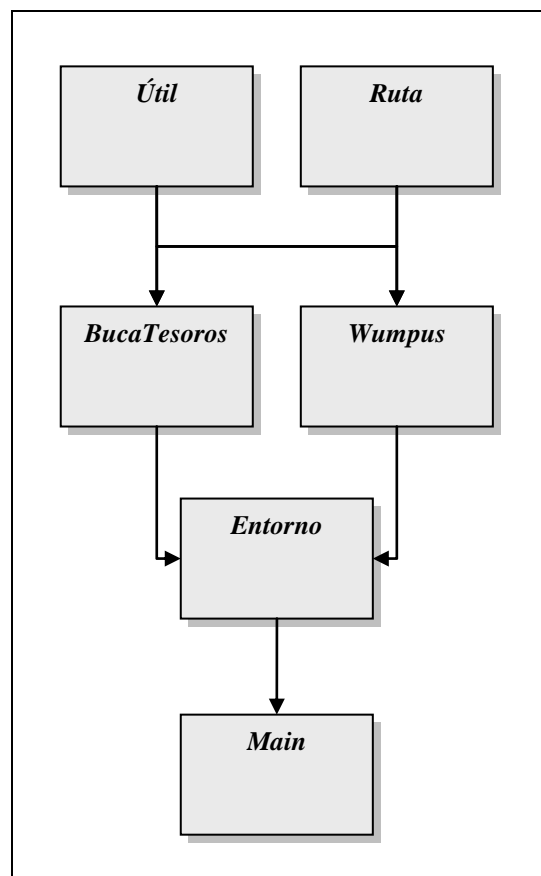


Diagrama 1. Diagrama de la estructura del programa.

### 2.1 Asesino

Activando esta opción le indicamos al monstruo que si el agente *BuscaTesoros* se encuentra a una cierta distancia se dirija hacia la posición del agente con el objetivo de matarlo. El atributo *radioOido* indica a que distancia es capaz

- *InfoBuscaTesoros*, muestra la información de los agentes

```
; variables de configuración
nombre                ; nombre del BuscaTesoros
posicion              ; posición del BuscaTesoros (x y)
(forma 'B)            ; pinta del BuscaTesoros
(direccion ESTE)      ; Norte, Sur, Oeste, Este
(numFlechas 1)        ; número de flechas disponibles
(asesino 1.0)         ; probabilidad de que el
                      ; BuscaTesoros dispare una flecha
(valiente 0.0)        ; probabilidad de atravesar una zona
                      ; peligrosa
(espontaneo 0.0)      ; probabilidad de generar una acción
                      ; espontánea
(metodoExploracion 'Aleatorio) ; indica cual es la estrategia que
                      ; utiliza el agente para buscar el
                      ; tesoro
                      ; Valores: Aleatorio
(mapaExploracion 'Si) ; usa el mapa durante la exploración
(metodoSalida 'DistanciaMinimaA) ; indica cual es método utilizado
                      ; por el agente para salir del
                      ; laberinto
                      ; Valores: Aleatorio,
                      ; DistanciaMinima, DistanciaMinimaA
(mapaSalir 'Si)       ; el agente crea un mapa y lo usa
                      ; para salir de la cueva
(tesorosBuscar 1)     ; le indica al BuscaTesoros el
                      ; número de tesoros que debe buscar
                      ; para poder abandonar la cueva
(maxTurnosBusqueda 50000) ; máximo número de turnos que se
                      ; pueden gastar en la búsqueda
(anchoMapa 15)        ; ancho máximo del mapa
(altoMapa 15)         ; alto máximo del mapa
```

### Código 1. Atributos del Agente

el monstruo de oír los movimientos de agente. Si el monstruo posee una capacidad de movimiento alta y usa esta estrategia, el monstruo suele acabar con el agente dado que esta no tiene posibilidad de huir. El *Mapa09.lsp* ilustra lo anteriormente expuesto.

- *InfoWumpus*, muestra la información del Wumpus

Si ponemos el campo *info* a *nil* se mostraran todos los campos de información.

## 3 Cambio de la interfaz de usuario

Para facilitar las pruebas hemos añadido una modificación que afecta a la forma de llamar a la función principal *main-run*. Esta función posee ahora un parámetro más: *info*, que contiene una lista con los elementos que queremos que se visualicen durante la simulación. Los valores posibles son:

- *Mapa*, muestra el mapa

## 4 Corrección de bugs

Tras realizar unas cuantas simulaciones nos dimos cuenta de que en determinadas ocasiones el agente se precipitaba contra los precipicios tras chocar contra un muro. Para arreglar el error sólo tuvimos que cambiar el orden en que el agente responde a los sensores. El orden actual es:

```
(BT-Test-MalOlor)
(BT-Test-Golpe)
(BT-Test-Brisa)
```

```
(BT-Test-Espontaneo)
(BT-Test-Salida)
(BT-Test-Grito)
(BT-Test-Brillo)
```

## 5 Más mapas

Para probar las nuevas características hemos añadido 5 nuevos mapas. Estos 5 junto con los 7 anteriores forman un total de 12 donde se prueban multitud de situaciones con diferentes algoritmos tanto para el agente *BuscaTesoros* como para el monstruo *Wumpus*. Al final de este documento se adjunta una copia de cada mapa.

```
nombre           ; nombre del Wumpus
posicion          ; posición del Wumpus (x y)
(forma 'W)        ; pinta del BuscaTesoros
(direccion NORTE) ; Norte, Sur, Oeste, Este
(movimiento 0.15) ; probabilidad de realizar un movimiento (0.0
                  ; => no se mueve nunca, 1.0 => se mueve
                  ; siempre)
(instinto 'Aleatorio) ; indica como de comporta el Wumpus. Valores:
                    ; Aleatorio, Territorial, Asesino
(radioOido 4)      ; indica el número de casillas en las cuales
                    ; el Wumpus es capaz de oír al BuscaTesoros
(radioTerritorio 3) ; radio del territorio del Wumpus
```

**Código 2.** Atributos del Wumpus